Background
Genome-wide association studies (GWAS) and phenome-wide association studies (PheWAS) involving one million GWAS samples from dozens of population-based biobanks present a considerable computational challenge and are carried out by large scientific groups under great expenditure of time and personnel. Automating these processes requires highly efficient and scalable methods and software, but so far there is no workflow solution to easily process one million GWAS samples.

Results
Here we present BIGwas, a portable, fully automated QC and association testing pipeline for large-scale binary and quantitative trait GWAS data provided by biobank resources. By using Nextflow workflow and Singularity software container technology, BIGwas performs resource-efficient and reproducible analyses on a local computer or any high performance compute system (HPC) with just one command, with no need to manually install a software execution environment or various software packages. For a single-command GWAS analysis with 974,818 individuals and 92 million genetic markers, BIGwas takes around 16 days on a small HPC system with only seven compute nodes to perform a complete GWAS QC and association analysis protocol. Our dynamic parallelization approach enables shorter runtimes for large HPCs.

Conclusions
Researchers without extensive bioinformatics knowledge and with few computer resources can use BIGwas to perform multi-cohort GWAS with one million GWAS samples and, if desired, use it to build their own (genome-wide) PheWAS resource. BIGwas is freely available for download from http://github.com/ikmb/gwas-qc and http://github.com/ikmb/gwas-assoc.
Reviewer Comments:

Reviewer 1

In this paper, the authors present an automatic quality control pipeline for genotyped data using the nextflow language, making it easier to perform large scale genome wide association studies. This is a welcoming addition to the field, simplifying the usual tedious and oftentimes complicated quality control procedures.

We thank the Reviewer for the supportive feedback and interest in our implementation. On our official GitHub tutorial websites https://github.com/ikmb/gwas-qc/ and https://github.com/ikmb/gwas-assoc, we have updated our software and documented the new software features in the Master Branch for the review process.

However, as of now, I was unable to run the proposed pipeline on the provided example dataset, and thus unable to provide an in-depth review on the performance / validity of the pipeline at the current stage.

We regret this and would like to know at which point in the instructions the problems occurred. For both the QC Module and Association Module, we have set up a dedicated Github webpage which, among other things, has a "Quick Start" guide at the beginning (https://github.com/ikmb/gwas-qc and https://github.com/ikmb/gwas-assoc) consisting of four short steps: (1) download the sample dataset (2) unpack the dataset, (3) QC Module execution with the example data set, which automatically installs the pipeline locally, (4) optional: execution of the Association Module using the example results of the QC Module and a direct link to the Assoc Module with more information about changing default parameters.

We would appreciate if reviewer #1 could run the “Quick Start” guide once and let us know if there may have been any technical issues with starting the pipeline.

Below are some of my comments based on the content of the paper and also my experience in trying to use the pipeline:

1. Under the quality control module, the authors suggested that all markers are matched with known variant database beforehand to bring the data to the same marker content. However, wouldn't this remove all customized markers?

We agree with the reviewer that matching variants with known marker ID from external databases is an important issue. In our implementation, variants of a custom chip are not lost during the matching step. The following sentence from the manuscript, section "Matching of Variant Identifiers with External Databases", already describes how we handle custom-array variants without existing marker IDs in external databases: "If no match could be found in the database, the original variant name is retained and prefixed with “unk_” to indicate that neither name, position, alleles nor strand alignment could be verified, which is often the case if the content is custom chip content.”
2. Is it really necessary to perform the HWE test on all permutation of batch combinations? Would it be enough to perform HWE in each batch once, then the whole data set once? If we only performed once in each batch, then once in the whole dataset, then this shouldn't be too computationally intensive.

We apologize if we have been unclear about this. The (admittedly short) answer is already given in the following sentence of the manuscript: “Batch-wise p-values are reported to find variants that fail HWE, among others, in a single control batch (but not in other control batches), in two or more control batches and in the whole collection of controls.”

What we do is that we test for deviations from HWE for each individual SNP variant in three ways: (a) HWE in each control batch individually, (b) in the entire control data set, and (c) in the entire control data set without one of the control batches. Thus, the number of HWE tests increases linearly with the number of control batches, and we are not performing a permutation of all control batches, which would indeed be very time-consuming. We then look for individual variants that pass the HWE P-value significance threshold in one of the following three ways: (i) the whole control dataset shows deviation from HWE, (ii) two or more control batches each show deviation from HWE, (iii) the whole control dataset without any of the control batches shows deviation from HWE. In all three cases, the variant is removed from the dataset. However, if case (iii) shows that after removing the control batch with the smallest (significant) HWE P-value, the total HWE P-value is no longer significant, then only the genotypes of the control batch with the smallest (significant) HWE P-value are set to “Missing”. By doing so, we generally allow to keep variants that have been “badly geno-typed” in only a single control batch.

3. Based on the content of the paper, it seems like this pipeline is specifically designed for case control phenotypes in European samples? Is it possible to extend the pipeline to continuous trait and also for non-European samples?

With the help of our colleague Dr. Lars Wienbrandt we have extended the quality control module for an analysis of quantitative traits. Therefore, Lars Wienbrandt is now also co-author on the manuscript. To clarify this, we now write in the Abstract: “Here we present BIGwas, a portable, fully automated QC and association testing pipeline for large-scale binary and quantitative trait GWAS data provided by biobank resources”.

Now the entire pipeline (QC module and association module) works for binary as well as quantitative traits. For a QC based on quantitative traits, in the PLINK input Fam file, only the 6th column must be specified as a quantitative trait (for more information, see also http://zzz.bwh.harvard.edu/plink/data.shtml#ped). We added this information to our tutorial website https://github.com/ikmb/gwas-assoc. This means that, as implemented in the PLINK program, it is assumed to be a quantitative trait as soon as values different from 1 (control), 2 (case) or 0 / -9 (missing) are defined in the 6th column of the PLINK Fam file. In this case, the HWE test is performed on all individuals (instead of only on the controls of a case-control study). In the Assoc module only the parameter --trait has to be set to “quantitative” instead of “binary” (see https://github.com/ikmb/gwas-assoc, section “Parameters”). PLINK as well as SAIGE will then automatically perform an association analysis for a quantitative trait, that means, in the association analysis a linear regression and a linear mixed model association analysis are performed instead of a logistic regression and a logistic mixed model association analysis. The information above is briefly described in sections “Quality Control Module” and “Association Testing Module”.
In principle, the QC module can be used for any European and non-European population. However, for the special case of an intended trans-ancestry GWAS study (for example to analyze GWAS samples of European, Asian and African descent), we recommend that a separate quality control be performed for each ethnic group. We added this information in the manuscript at the end of section “Quality Control Module”. The "ethnicity_predicted" column in the sample annotation file (see tutorial on https://github.com/ikmb/gwas-qc) is there, first, to help the user remember which ethnicity to assign the study participants to. Then, based on the numerically most frequently specified ethnicity in column "ethnicity_predicted" the HWE test is performed only for controls of that ethnicity (or for all samples with that ethnicity if it is a quantitative trait). To better emphasize this, we added the following sentence to our QC tutorial website: “Based on the numerically most frequently specified ethnicity in column "ethnicity_predicted" the HWE test is performed only for controls of that ethnicity (or for all samples with that ethnicity if it is a quantitative trait).”

We further added the following sentence to section “Test for Hardy-Weinberg-Equilibrium” in the manuscript: “Hardy-Weinberg equilibrium tests are performed for controls only in a case-control study (and for all individuals if a quantitative trait is to be considered).”

Afterwards, based on the total GWAS dataset, the PCA analysis with the 1000 Genomes reference samples removes study samples (per default) which have values smaller than the median plus/minus five times the interquartile range (median ±5×IQR; parameter adjustable) of all GWAS study samples for the first two PCs. To better emphasize this, we adjusted the following sentence in section “Controlling for population stratification and other confounders” in the following way: “Individual GWAS samples (grey crosses) with values smaller than the median plus/minus five times the interquartile range (median ±5×IQR; samples outside the red square in PCA plot) represent PCA outliers for the first two PCs and are automatically removed (per default; adjustable).” For example, if a sufficiently large number of European and non-European origin GWAS samples are in the dataset, then this approach will result in no outliers being detected. However, if there are only a small number of non-European GWAS samples in a large dataset of European GWAS samples (or vice versa), this will have the desired effect of detecting the non-European GWAS samples as PCA outliers in the dataset of European GWAS samples (or vice versa). We recommend that the user considers the sample selection regarding the ethnicity in advance of the GWAS study, because even the best automated QC pipeline cannot replace a coherent study design.

Also, the Association Module is not specialized for individuals with European ancestry and can be used for cohorts of any ancestry as long as they represent a reasonably homogeneous study population. However, for the special case of a trans-ancestry association analysis, we recommend either performing a single association run for each ethnic group or using specific trans-ancestry association testing programs, for example, MANTRA (PubmedID 22125221) or modified random effect model (RE2C; PubmedID 28881976), that account for differences in allelic effects resulting from differences in linkage disequilibrium (LD) between (the user-defined) populations. To make this clear, we’ve already written in the Discussion section that “BIGwas is not yet suitable for GWAS analysis of family data or GWAS analysis across populations of worldwide ethnicities.”

4. Will pruning be performed before performing the heterozygosity check?

The distribution of mean heterozygosity (excluding sex chromosomes) across all individuals is calculated based on all quality-controlled genetic markers that passed “SNP QC I” and does not include an additional pruning step.

5. Will the same 14,484 SNPs always be used for IBS and IBD estimation? If not, then it might be better to describe the pruning procedure as the 14,484 number is likely...
specific to the dataset used by the author.

In another test run for the so-called Immunochip (not mentioned in the manuscript) pruning ended up with 14,484 SNPs as the best suitable set of independent SNPs for PCA and the “Duplicate/Relatedness Detection” analysis. We apologize for this misleading statement in the manuscript and removed this number. Instead, the “Duplicate/Relatedness Detection” analysis uses the same set of variants that is dynamically determined for the PCA analysis as described in this sentence: “For all PCA, a set of independent (MAF >0.05) SNPs is calculated, excluding non-autosomes, variants in LD (leaving no pairs with r2>0.2, within 50 kb windows) or within the extended major histocompatibility complex (xMHC; chr6:25-34Mb), and 11 high-LD regions as described by Price et. al. [31].” We have revised the following text passage about IBS/IBD estimation as follows: “For robust duplicate/relatedness testing (identity-by-state (IBS) and identity-by-descent (IBD) estimation), we use the same set of independent variants, which was dynamically determined for PCA analysis (see above).”

6. Will the pipeline filter out samples with mismatch self-reported and genetically indicated sex information? And will the pipeline remove samples with sex aneuploidy?

No, the QC module will not filter out samples where there may be a discrepancy between self-reported and genetically determined biological sex. The reason for this is that common sex-detection methods, such as the method used by PLINK (http://zzz.bwh.harvard.edu/plink/summary.shtml#sexcheck), only use chromosome X variants and therefore, in our experience, do not work well in many individuals.

We rather recommend a verification between self-reported and genetically determined biological sex by means of normalized signal intensities of SNPs on chromosome X as well as chromosome Y simultaneously. However, because this step requires e.g. Illumina iDat files raw data with normalized signal intensities instead of genotypes in PLINK format, we have omitted this step in the QC Module. To clarify this, we have included the following sentence in the Methods section: “The QC module does not perform a check between self-reported and genetically determined biological sex based on chromosome X PLINK genotypes; we recommend using normalized signal intensities of SNPs on chromosome X as well as Y for this purpose.”

The PLINK genotype and vcf files from SNP arrays and/or genotype imputation processes do not provide information about an abnormal number of sex chromosomes. For this reason, individuals with sex chromosome aneuploidies cannot be detected.

7. It is possible to convert BGEN data to VCF using PLINK2, which also natively support BGEN.

It is true that the UK Biobank’s latest GWAS data release contains, in addition to unimputed genotype data in PLINK format, imputed data in the BGEN format instead of the much more common VCF format. However, since the more common VCF format is a generic format used by biobanks and genotype imputation programs (TOPMed and Sanger Imputation server output, among others) worldwide, and because the conversion from BGEN to VCF took an extremely long time and cost an extremely large amount of hard disk space with UK Biobank’s cohort size for our benchmark purposes, we chose to drop UK Biobank’s BGEN support by default. Thus, in addition to SAIGE, we can continue to use PLINK as the de facto gold standard tool for regression-based association analyses in our Association Module, even for the analysis of UK Biobank, because PLINK cannot directly read BGEN for association
analyses. Because UK Biobank is, to our knowledge, the only dataset where the BGEN format is used by default, we offer a simplified the association pipeline for the UB Biobank case and a set of scripts to perform SAIGE association testing directly on UKB BGEN files on https://github.com/ikmb/gwas-assoc, see section “UK Biobank Proof-of-concept”.

8. With relatedness filtering, can the pipeline automatically calculate the full relatedness matrix? E.g. for UK Biobank, the full 500,000 x 500,000 matrix? Or does it perform filtering within each 2,000-sample cluster, then merge the post-filtering sample and repeat until no-relatedness remain? In the latter case, how do you ensure the maximum number of samples are retained? Is ‘--rel-cutoff’ used?

Indeed, our QC Module calculates the full relatedness matrix from all 488k UK Biobank samples (or even from 976k samples, for benchmark results see Table 2) without using Plink’s ‘--rel-cutoff’ (i.e. there is no exclusion of one member of each pair of samples with observed genomic relatedness greater than a given cutoff value). Instead of using n-sample clusters for parallelization, we used PLINK’s IBD interface using ‘--genome’ for calculating the full relatedness matrix and run all combinations of sample pairs in parallel chunks using different compute jobs. The number of jobs running in parallel is dynamically selected based on the number of samples (see also Figure 2).

9. It is rather surprising that the whole pipeline requires 7 days to complete. Which is the most time-consuming step? It’d be interesting to know. I’d imagine the PCA and Relatedness matrix require the most time to compute.

We are of the opinion that a complete QC and association analysis in only 7 days is unusually fast for a GWAS dataset of size 974,818 samples and 92 million variants with a limited use of 150 CPU cores for computation. This is actually possible because we dynamically parallelized as many steps of QC and association analysis as possible with respect to the number of samples (n) and variants (m; see also Figure 2 and 3).

In terms of CPU time, most CPU hours are consumed during our extensive Hardy-Weinberg analysis in the part where we re-calculate the HWE for each leave-one-batch-out permutation. Although this part is highly parallelized, it still accounts for a large fraction of the total time because we limit the number of active parallel jobs on our high-performance cluster (HPC) to 150 (by default; but configurable). Another time-consuming step is merging the input datasets with reference genotype data sets in preparation for PCA, i.e., merging the input dataset with the HapMap and 1000 Genomes files, which is not parallelizable by default with Plink 1.9 and is not currently available in Plink 2. Interestingly, the PCA and relatedness matrix computation consumes less than 20% of the wall clock time. One reason for this is the fact that we use the FlashPCA2 program for PCA (Abraham et al., 2017, PubMedID 28475694), which can perform PCA on 1 million individuals faster than other competing approaches, while requiring substantially less memory (see also section “Sample QC” in our manuscript).

Below are some comments regarding the implementation of the pipeline:

1. As of now, running the pipeline require extensive knowledge of the nextflow language and behavior. For example, any cluster options written in the local *.config files are going to be overwritten by the $HOME/.nextflow/assets/ikmb/gwas-qc/nextflow.config file (which has process.executor predefined as slurm).
We thank the Reviewer for pointing out this issue with the process executor variable of the Slurm batch system. Users do not need much experience with the Nextflow programming environment to start our pipeline and this default setting of the Slurm batch system was not known to us until now. We have now removed the setting for this process executor variable in our current source code so that the pipeline runs locally by default. The process for setting the correct executor for the HPC system is now explained in the readme file, see section “UKSH medcluster configuration”. This change must be done in the nextflow config file which is located at $HOME/.nextflow/config.

2. On some HPC, the $HOME directory can have limited space (e.g. one from my institute only have 20GB). Given that the current pipeline use $HOME for temporary output, users can quickly run out of spaces. To avoid that, users will need to do export _JAVA_OPTIONS=-Djava.io.tmpdir=<some folder with more storage>

We assume that this is a misunderstanding because no change needs to be made to the Java installation. In Nextflow it is quite easy to set the NXF_WORK environment variable to a location of your choice (https://www.nextflow.io/docs/latest/config.html). Variable NXF_WORK points to a user defined directory where working files are stored (usually your scratch directory). We thank the Re-viewer for this advice and have updated our readme online documentation to explain where to set the Nextflow working folder.

3. When I tried to run the pipeline, I cannot get pass the RS.nf stage, which gives me an er-ror: FATAL: Cached File Hash(sha256.33fc75ba032f40e5df1292914c250482a4febe34d274bea995bcaba0b82c3befc) and Expected Hash(sha256.707eb1d6fd5b7b93e01f6bfa3fffc4a7efb62541d8194b815d9359d9510091) does not match

We regret the occurrence of this error message and we were also wondering how such an error message can occur. We have investigated different Nextflow versions and user forums and it seems like that there is a problem in Nextflow that is (very rarely) triggered when the directory in $NXF_WORK or the Java tempdir with the above _JAVA_OPTIONS is set to a Samba/CIFS net-work share or sometimes to a poorly configured service that introduce file caching incoherence (deprecated NFS, misconfigured Lustre mounts, S3 mounts). We have now included a few worka-round solutions on how to set up $NXF_WORK on a SAMBA shared directory in our readme online documentation, see section “Cache issues”. Nevertheless, we do not recommend working on a SAMBA directory when specifying $NXF_WORK.

4. One disadvantage of using nextflow is its inability of deleting intermediate files. Deleting the intermediate files will invalidate -resume, requiring users to restart the whole pipeline from scratch instead of picking up from where they end. Within the current pipeline, there are around 50+ in-stance of `--make-bed`, which generates large bed files. Instead of using `--make-bed`, it might be better to use `--make-just-fam` and `--write-snplist` in place of `--make-bed` whenever possible such that only the ID of SNPs and samples were reported between steps instead of writing out the whole genotype matrix. `--extract`, `--exclude`, `--keep` and `--remove` can then be used to per-form cheap on the fly filtering on subsequent steps. This should also significantly reduce IO burden and thus improve performance. However, it is acknowledged that `--make-bed` is still required for operations such as merging the different batch.
We thank the reviewer for these good suggestions for changes. Now we have switched to these lightweight options where possible, resulting in a 20% reduction in intermediate file size for our example dataset. We are very happy about that. However, there are still a few instances of --make-bed where the intermediate genotype matrix is used as input for non-PLINK tasks without the ability to use exclude/keep lists. Further, some time-consuming steps such as merging PLINK files before PCA remain and unfortunately cannot be bypassed. Interestingly, in terms of runtime, we have not yet been able to detect any measurable runtime reductions as a result of the changes, which may also be due to our fast file storage system.

5. Given the design of the current pipeline, it might be beneficial to use the latest DSL 2.0 feature of nextflow, which allow piping and modulating different processes. This should substantially improve the organization of the scripts, avoid the need to calling another nextflow instance within the current nextflow pipeline.

We agree that the latest DSL 2.0 would potentially be beneficial to the overall structure of the pipeline to make future processes easier to modify. However, this pipeline is the result of a lot of implementations we have done over the last few years and already existed at its core when DSL 2.0 was first introduced as a first stable release in June 2020. A completely new pipeline would certainly use DSL 2.0, but the DSL version is not critical for the user. Re-implementing the pipeline in DSL 2.0 with the typical several rounds for debugging would require an enormous amount of time without any apparent benefit we see for the user.

6. It might be beneficial to provide a command line interface (e.g. utilize the $param object from nextflow) so that users can easily specify parameters with `--output`, `--datasets` etc.

Since our pipeline supports multiple datasets, where each dataset can contain multiple batches, we make a lot of use of arrays in our configuration files. Unfortunately, Nextflow does not support specifying array contents on the command-line. However, it is possible to specify simple (i.e., non-array) parameters on the command-line that would otherwise appear in the pipeline.config file, in particular the `--output` parameter and the `--qc_config` parameter.

Reviewer 2

Drs. Kässens and Ellinghaus present BIGwas, a single-command analysis tool to jointly run data quality control and association analysis in the context of genome-wide association studies (GWAS). The Authors should be commended for dedicating time and releasing such a software, as there is great need of standardized quality control procedures in the GWAS field.

We thank the Reviewer for the supportive feedback and interest in our software.

On our official GitHub tutorial websites https://github.com/ikmb/gwas-qc/ and https://github.com/ikmb/gwas-assoc, we have updated our software and documented the new software features in the Master Branch for the review process.

I would like to point the authors' attention towards some points that, in my opinion,
deserve some clarification.

- When I started reading the manuscript it took me a while before I realised that this is a tool for quality control from the individual data of each participant. Perhaps I was misled by the fact that many GWAS quality control softwares only focus on post-GWAS results. Perhaps this deserves immediate clarification at the beginning of the paper.

We thank the reviewer for this valuable comment. To further clarify this point, we have included the following underlined text at the end of the introduction: “… there are still no easily executable and reproducible software pipelines available for performing large-scale quality control (QC) and association tests on a large number of individual (large-scale) GWAS or PheWAS data sets totaling mil-lions of GWAS samples.

- Linked to the previous discussion, I would ask the authors to put the use of BIGwas a bit better into the context of consortia that include hundreds of studies, some very large like UKBB, others very small (1000-2000 samples): is there a real advantage to using BIGwas, given that many stud-ies have their pipelines already implemented and many are not willing to change them? What might be the advantages/disadvantages?

We agree with the reviewer that BIGwas best realized to its full potential in the context of a mega-GWAS study, where the user has to perform quality control (QC) analysis across dozens of co-horts simultaneously. To better emphasize this, we have included the following underlined sentence at the end of the introduction: “In particular, parallel and consistent processing and analysis of doz-ens of individual GWAS cohorts in the context of a mega-GWAS analysis is enormously time-consuming and can quickly become unmanageable if processed manually.” In addition, we have modified the title to read: “BIGwas- Single-command Quality Control and Association Testing for multi-cohort and biobank-scale GWAS/PheWAS data.”

On the other hand, BIGwas also demonstrates its full potential in the execution of single-cohort GWAS studies: Due to the Singularity container and Nextflow pipeline technology, extensive installa-tion is not required and only a single command is needed to perform a complete QC and/or associa-tion analysis. This makes BIGwas thus also interesting for users who may only want to perform a single-cohort GWAS study only once as part of a larger multi-omics study. This saves these users lots of training time to acquire knowledge in the field of genome-wide association studies (GWAS).

- Since BIGwas is implemented in the context of GWAS consortia, in my opinion it would be appro-priate not only to compare it with H3Agwas but also with mixed approaches (e.g. study-specific pipelines for post-GWAS analysis and quality control, eg GWAToolbox or similar). If a formal com-parison is not feasible, perhaps one could at least discuss advantages/disadvantages of alternative approaches.

Due to the above-mentioned reasons, we believe that BIGwas can show its potential in both meg-agGWAS and single-cohort GWAS analyses. We also understand well that there is a desire for a comprehensive comparison of BIGwas with existing GWAS analysis packages in R that can be used to manually perform QC or GWAS association analysis steps. However, we would then turn our comparison to existing software programs and packages, with which one can execute certain subtasks of the analysis pipeline only by a manual call of certain functions. The intention of our pipe-line development was to allow QC and Assoc testing to be performed in a standardized and repro-du-cible manner with a single command, and for users new to GWAS to not have to spend a lot of time dealing with a variety of software packages when the goal is
to easily perform a state-of-the-art GWAS analysis.

We agree that BIGwas may not be of as much interest to the very experienced GWAS user who wants full control over all possible packages and analysis steps and wants to build her/his own pipeline. Therefore, in order to highlight the advantages and disadvantages of using BIGwas in comparison to existing R packages we have added the following sentence to the Discussion section: “BIGwas differs from existing GWAS data processing packages in R in that it allows standardized, reproducible, and high-throughput processing of GWAS datasets via a single command, whereas existing GWAS R packages allow for more flexible processing, for example, with respect to the order of processing steps.”

- It seems that a limitation of BIGwas is the very strong reference to the IBDGC and Severe Covid-19 GWAS group consortia’s protocols. This affects many aspects of the proposed analysis plan. For example, it is very common to find groups that do not use PLINK at all, neither at the quality control level nor at the analysis level; how should such groups behave in the management of e.g. the X chromosome or other steps if they wanted to use BIGwas?

Now we have removed the reference to the IIBDGC Consortium and the GWAS Group in the abstract, as the QC and association testing protocol applied is not only applicable to the analysis of diseases from these alliances but can be applied to various binary and quantitative traits in general. We agree that mentioning these disease-specific alliances in the Abstract may confuse the reader.

We cannot rule out the possibility that there are very experienced GWAS analysts who prefer R-only packages, already have own their own software pipeline, and want to avoid PLINK entirely. On the other hand, we see no genuine reason not to use PLINK for specific subtasks such as IBD/IBD estimation, as it is the most widely used and robust C++ software tool available for GWAS studies, with an excellent code base and extensive use of bit-level parallelism (Chang et al., Gigascience. 2015 Feb 25;4:7). We still think we can convince even the very experienced GWAS analysts to use BIGwas because, as noted, we offer a complete QC and association analysis including association analysis for the X chromosomal regions via a single command. If the QC module of BIGwas is not of interest, the very experienced GWAS analyst might still be interested in the Association Module of BIGwas. The User could then easily use only the Association Module of BIGwas, since the two modules of BIGwas run completely independently of each other.

- Apparently, the pipeline is thought for binary trait analysis (cases and controls). Is it also suitable for quantitative traits? Can the Authors specify this better?

With the help of our colleague Dr. Lars Wienbrandt we have extended the quality control module for an analysis of quantitative traits. Therefore, Lars Wienbrandt is now also co-author on the manuscript. To clarify this, we now write in the Abstract: “Here we present BIGwas, a portable, fully automated QC and association testing pipeline for large-scale binary and quantitative trait GWAS data provided by biobank resources.”

Now the entire pipeline (QC module and association module) works for binary as well as quantitative traits. For a QC based on quantitative traits, in the PLINK input Fam file, only the 6th column must be specified as a quantitative trait (for more information, see also http://zzz.bwh.harvard.edu/plink/data.shtml#ped). We added this information to our tutorial website https://github.com/ikmb/gwas-assoc. This means that, as implemented in the PLINK program, it is assumed to be a quantitative trait as soon as values different from 1 (control), 2 (case) or 0 / -9 (missing) are defined in the 6th column of the PLINK Fam file. In this case, the HWE test is performed on all individuals.
In the Assoc module only the parameter --trait has to be set to "quantitative" instead of "binary" (see https://github.com/ikmb/gwas-assoc, section "Parameters"). PLINK as well as SAIGE will then automatically perform an association analysis for a quantitative trait, that means, in the association analysis a linear regression and a linear mixed model association analysis are performed instead of a logistic regression and a logistic mixed model association analysis. The information above is briefly described in sections "" and "Association Testing Module".

- It is unclear whether the IBD QC step will exclude related individuals or if related individuals can be retained in the analysis by using linear mixed modeling approaches. These models could also incorporate a random batch effect, for instance, thus making the batch effect QC step unnecessary.

We thank the reviewer for this valuable advice. Per default we determine the relatives using IBD/IBS analysis, remove them from the final dataset, and output the list of relatives in an additional text file during SampleQC so that the user could add them back later if desired. For the QC Module, we have now implemented an additional switch "--keep_related" which, if set, keeps the relatives in the final output files from QC. This parameter can be specified in the dataset-specific configuration, the general pipeline configuration, or directly on the command line. For the Association Module, the user can still decide by specifying the samples in the fam file, which samples she/he wants to use for the association analysis.

- In addition, to PLINK and SAIGE, there are other popular GWAS software such as BOLT-LMM, REGENIE, and others. Can they be implemented within the BIGwas analysis pipeline?

In principle, yes. We decided not to add BOLT-LMM (purely linear mixed model without providing a logistic mixed model for binary traits) in addition to PLINK logistic/linear regression and SAIGE logistic/linear mixed model analysis in our Association Module because, on the one hand, it further increases the overall runtime as well as the number of jobs running in parallel, and, on the other hand, we don't expect too much gain from an additional linear mixed model tool such as BOLT-LMM, as the developers of BOLT-LMM themselves state the following on their website: "We recommend BOLT-LMM for analyses of human genetic data sets containing more than 5,000 samples. The algorithms used in BOLT-LMM rely on approximations that hold only at large sample sizes and have been tested only in human data sets. For analyses of fewer than 5,000 samples, we recommend the GCTA or GEMMA software. We also note that BOLT-LMM association test statistics are valid for quantitative traits and for (reasonably) balanced case-control traits. For unbalanced case-control traits, we recommend the SAIGE software."

REGENIE is an interesting new tool recently published on bioRxiv. REGENIE allows parallel association analysis of multiple phenotypes simultaneously from a single input dataset such as UK Biobank, which is particularly interesting when each association analysis always refers to the same GWAS input dataset and only the phenotype information changes per run. This type of analysis requires a range of phenotypes for individuals in the same dataset, as is the case with UK Biobank. Therefore, we see REGENIE as a special tool that we could add in the future as an additional association Nextflow module for multi-phenotype analysis based on a single GWAS input dataset.

- GWAS and PheWAS are discussed under the same umbrella; however, the whole pipeline is presented and discussed only with regard to GWAS; please outline key
We agree with the reviewer that the focus of the benchmark here was to demonstrate the simplicity and reproducibility of a multi-cohort mega-GWAS QC and large-scale association analysis. Now it is not a far step to create a PheWAS resource (e.g., as provided by the Michigan Genomics Initiative (MGI), see http://pheweb.sph.umich.edu/) based on the genetic data when numerous different phenotype information are available for the same individuals. For example, to perform a (genome-wide) PheWAS analysis with five different traits for one and the same data set, it is sufficient to create five different PLINK fam files for binary or quantitative traits for the same GWAS input data set and to call the pipeline five times each by a single command.

To better highlight this PheWAS usability, which now offers itself following the GWAS analyses with Bigwas, we have included the following sentence in the Introduction section about existing PheWAS resources: “To create a (genome-wide) PheWAS resource of this type, Bigwas can be used alongside the QC and mega GWAS analysis capability to test for association with a wide variety of traits in parallel or sequentially (requiring only one single program call per trait on a HPC) when numerous different phenotype information is available for the same individuals.”

- Please, explain all jargon. For instance, most audience might be unfamiliar with terms such as “containerized”.

We apologize for assuming the term “containerized” to be familiar to those reading the manuscript. We have revised the Abstract and the first sentence of the Discussion as follows: “Here we present Bigwas, a portable, fully automated QC and association testing pipeline for large-scale GWAS data provided by biobank resources.” and “We have introduced Bigwas, a portable (i.e. Singularity container-based), fully automated GWAS quality control (QC) and association pipelines in Nextflow that can be executed with a single command without manual installation by the user.

- In the Abstract, I would suggest to keep the approach general, avoiding referring to specific situations that might soon become outdated (e.g., “the number of GWAS samples is twice as large as the current GWAS data release of the UK Biobank”.

We thank the reviewer for this advice and have revised this sentence in the abstract as follows: “For a GWAS analysis with 974,818 individuals and 92 million genetic markers, Bigwas takes around 16 days on a small high performance compute system (HPC) with 7 compute nodes to perform a complete GWAS QC and association analysis protocol.”

- Please, review text for typos. For instance, I found “per centage” in place of “percentage”.

We have corrected this and other typos.
| special series or article collection? | Yes |
|---|---|
| **Experimental design and statistics** | Yes |
| Full details of the experimental design and statistical methods used should be given in the Methods section, as detailed in our [Minimum Standards Reporting Checklist](https://example.com). Information essential to interpreting the data presented should be made available in the figure legends. | |
| Have you included all the information requested in your manuscript? | |

| **Resources** | Yes |
|---|---|
| A description of all resources used, including antibodies, cell lines, animals and software tools, with enough information to allow them to be uniquely identified, should be included in the Methods section. Authors are strongly encouraged to cite [Research Resource Identifiers (RRIDs)](https://example.com) for antibodies, model organisms and tools, where possible. | |
| Have you included the information requested as detailed in our [Minimum Standards Reporting Checklist](https://example.com)? | |

| **Availability of data and materials** | Yes |
|---|---|
| All datasets and code on which the conclusions of the paper rely must be either included in your submission or deposited in [publicly available repositories](https://example.com) (where available and ethically appropriate), referencing such data using a unique identifier in the references and in the “Availability of Data and Materials” section of your manuscript. | |
| Have you have met the above requirement as detailed in our [Minimum Standards Reporting Checklist](https://example.com)? | |
BIGwas– Single-command Quality Control and Association Testing for multi-cohort and biobank-scale GWAS/PheWAS data

Jan Christian Kässens¹,²*, Lars Wienbrandt¹ and David Ellinghaus¹,³*

¹Institute of Clinical Molecular Biology, Christian-Albrechts-University of Kiel, Kiel, Germany and ²Haematology Lab Kiel, Klinik für Innere Medizin II, University Hospital Schleswig-Holstein, Germany and ³Novo Nordisk Foundation Center for Protein Research, Disease Systems Biology, Faculty of Health and Medical Sciences, University of Copenhagen, Copenhagen, Denmark

*Corresponding authors, janchristian.kaessens@uksh.de, d.ellinghaus@ikmb.uni-kiel.de

Abstract

**Background** Genome-wide association studies (GWAS) and phenome-wide association studies (PheWAS) involving one million GWAS samples from dozens of population-based biobanks present a considerable computational challenge and are carried out by large scientific groups under great expenditure of time and personnel. Automating these processes requires highly efficient and scalable methods and software, but so far there is no workflow solution to easily process one million GWAS samples.

**Results** Here we present BIGwas, a portable, fully automated QC and association testing pipeline for large-scale binary and quantitative trait GWAS data provided by biobank resources. By using Nextflow workflow and Singularity software container technology, BIGwas performs resource-efficient and reproducible analyses on a local computer or any high performance compute system (HPC) with just one command, with no need to manually install a software execution environment or various software packages. For a single-command GWAS analysis with 974,818 individuals and 92 million genetic markers, BIGwas takes around 16 days on a small HPC system with only seven compute nodes to perform a complete GWAS QC and association analysis protocol. Our dynamic parallelization approach enables shorter runtimes for large HPCs.

**Conclusions** Researchers without extensive bioinformatics knowledge and with few computer resources can use BIGwas to perform multi-cohort GWAS with one million GWAS samples and, if desired, use it to build their own (genome-wide) PheWAS resource. BIGwas is freely available for download from http://github.com/ikmb/gwas-qc and http://github.com/ikmb/gwas-assoc.

**Key words:** GWAS; PheWAS; biobank; quality control; association testing; pipeline; Nextflow; Singularity; scalability

Introduction

Genome-wide association studies (GWAS), in which millions of genetic variants are tested across the genomes of study individuals to identify genotype–phenotype associations, have revolutionized the field of complex disease genetics over the last ten years. Decreasing costs of genome-wide genotyping with SNP arrays allow routine collection of GWAS data from deeply phenotyped biobank population cohorts, where many clinical features can be assessed beyond disease status. For this reason, it is possible to conduct GWAS studies with over one million samples [1] using GWAS data from large biobanks.
BIGwas comprises two different modules, namely the Quality Control (QC) Module and the Association Testing Module, that are part of a typical GWAS workflow (Figure 1). While the

BIGwas, a fast and efficient platform-independent software pipeline for QC and association testing of large-scale GWAS/PheWAS data sets, which automatically processes one million GWAS samples. By using Nextflow workflow technology [14] in combination with Singularity software container technology [15] (Methods), BIGwas can be started with a single command (Figure 1), with no need to manually install a full software execution environment or various software packages. BIGwas implements a variety of best practice

Figure 1. Overview of the single-command quality control and association testing workflow of BIGwas. Multiple GWAS data sets in PLINK format from any genotyping platform can be used as input for the Quality Control (QC) Module. Both the Quality Control Module and the Association Testing Module can be started and executed with a single command. The genotype output files of Quality Control Module are used directly as input for genotype imputation programs such as the TOPMed Imputation Server [12]. The QC output files of the imputation process can be directly used as input for the Association Testing Module, which implements logistic/linear regression and mixed model association testing using PLINK [5] and SAIGE [13].
The Quality Control module works for binary as well as quantitative traits and is subdivided into five distinct phases: Pre-QC, SNP QC I, Sample QC, SNP QC II and Final Analysis and Report Generation. For a quality control based on quantitative traits, in the PLINK input Fam file, only the 6th column must be specified as a quantitative trait (for more information, see http://www.broad.harvard.edu/plink/data.shtml). In Pre-QC, an unlimited number of input GWAS data sets (batches) in PLINK format from different Affymetrix and Illumina platforms are, among other steps, matched with known variant databases from dbSNP and currently 140 Illumina/Affymetrix GWAS platforms and then brought to the same marker content. SNP QC I primarily filters variants based on missingness and Hardy–Weinberg equilibrium (in controls only) across and within batches (i.e. genotype files or self-defined batches) as well as batches from a particular phenotype and the entire control group with a user-defined FDR threshold. Finally, all information gathered through the QC pipeline is written to a single PDF report (see Figure 2). In principle, the QC module can be used for any European and non-European population. However, for the special case of an intended trans-ancestry GWAS study (for example to analyze GWAS samples of European, Asian and African descent), we recommend that a separate quality control be performed for each ethnic group.

**Methods**

**Quality Control Module**

The Quality Control module works for binary as well as quantitative traits and is subdivided into five distinct phases: Pre-QC, SNP QC I, Sample QC, SNP QC II and Final Analysis and Report Generation. For a quality control based on quantitative traits, in the PLINK input Fam file, only the 6th column must be specified as a quantitative trait (for more information, see http://www.broad.harvard.edu/plink/data.shtml). In Pre-QC, an unlimited number of input GWAS data sets (batches) in PLINK format from different Affymetrix and Illumina platforms are, among other steps, matched with known variant databases from dbSNP and currently 140 Illumina/Affymetrix GWAS platforms and then brought to the same marker content. SNP QC I primarily filters variants based on missingness and Hardy–Weinberg equilibrium (in controls only) across and within batches (i.e. genotype files or self-defined batches) as well as batches from a particular phenotype and the entire control group with a user-defined FDR threshold. Finally, all information gathered through the QC pipeline is written to a single PDF report (see Figure 2). In principle, the QC module can be used for any European and non-European population. However, for the special case of an intended trans-ancestry GWAS study (for example to analyze GWAS samples of European, Asian and African descent), we recommend that a separate quality control be performed for each ethnic group.

**Individual Annotations**

A batch affiliation can be defined independently from a single GWAS batch using the annotations file. In this file, separate batch identifiers can be assigned for the genotyping batch, country of origin and diagnosis (if non-control). Each of these identifiers will be used in principal component plots, histograms and ANOVA tests to identify batch effects within input data sets.

**Assignment of variants to Pseudo-autosomal Regions**

The first step consists of checking and assigning variants to the pseudoautosomal regions (PARs, which are inherited just like autosomal chromosomes) on chromosomes X and Y, as defined by genomic positions in the current genome builds. The genomic annotation of the PAR region have undergone a series of changes throughout different Plik versions. In earlier versions, Plik stored both PAR1 and PAR2 variants separate from the X (23) and Y (24) chromosomes, namely in a virtual chromosome 25, while newer Plink versions use alpha-numerical chromosome codes such as PAR1 and PAR2. As not every tool is aware of these changes, they are often mixed up in chromosomes 23, 24, 25, or all of them. The Pre-QC phase consistently moves both PAR1 and PAR2 variants back into chromosome 25 according to the coordinates published by the Genome Reference Consortium for GRCh37 [21]; after the QC, chromosome 25 variants are moved back to chr23 for the generation of the separate imputation vcf input files, since this is required by the TOPMed Imputation Server, see section on Variant Call Format Output below. If a data set presents heterozygous calls for males outside the PAR1/PAR2 regions, and hence, belong to the non-PAR regions on chromosomes X and Y, heterozygous genotype calls for males are likely to be sequencing errors and will be set to missing. The QC module does not perform a check between self-reported and genetically determined biological sex based on chromosome X PLINK genotypes; we recommend using normalized signal intensities of SNPs on chromosome X as well as Y for this purpose.

**Automatic Chip Type and Variant Annotation Detection**

Since many GWAS platforms (microarray or chip type) have specific properties and exhibit potential biases, it is crucial for the QC to know the exact chip type from which the data set was generated. This step helps identifying the chip type by

---

Figure 2. Workflow graph of the Quality Control (QC) Module of BiGWAS, with the default number of threads used in each process (for details see Methods). n denotes the number of parallel jobs or threads automatically scales with the number of samples in the data set.
matching chromosome IDs, variant IDs, base-pair positions and alleles of variants from PLINK’s bim files against a set of currently 140 original GWAS platform content information files (called strand files) provided by Illumina and Affymetrix [22]. The results of the chip detection step are expressed as match rates (in percentages) and can not only be used to verify the true chip type, but also whether variant identifiers and/or the string information have already been manipulated by the user. Our current chip detection database contains 140 different chip types (including genome builds 37, 38 and partially 36) resulting in 415 chip versions that can be detected. The strand files are processed by a configurable number of threads (per default 2 threads only) where each thread processes a strand file as follows: First, every file entry (i.e. variant) of PLINK’s bim file is matched against the strand file using variant ID, position and alleles. After the final variant match, a composite match rate is calculated to assess how well a particular strand file (or chip type) matches the annotation in the bim file. After each strand file is processed, the list of chip matches is sorted by a weighting formula, with $m$ denoting the respective match rate:

$$m_{\text{sort}} = m_{\text{ID}} \cdot m_{\text{pos}} \cdot \max \{m_{\text{orig_strand}}, m_{\text{plus_strand}}\}$$

It is often the case, that variants are converted to the plus strand before QC. To make the match sorting robust, both the allele match rate for the original strand definitions and the allele match rate for the plus strand will be calculated. The strand files are then sorted by $m_{\text{sort}}$ and can be used later for an optional genome build lift-over process (non–hg19 data and liftover to hg19). If the bim files match the plus strand better than the original strand information, a note is displayed in the PDF report. For performance reasons, the detection software has been written in the Rust programming language and is also available as a separate application (see Sect. Availability of Source Code and Requirements).

Automatic Genome Build Lift-over of Genomic Coordinates

Currently, the QC pipeline only supports data on genome builds 19 (or GRCh37). If input data is in a different format, e.g. hg18 (GRCh36) or hg38 (GRCh38), the lift–over step can be explicitly enabled to perform an automatic lift–over to hg19. If the chip type is known beforehand or from the automatic chip type detection step (see above), a lift–over request can be set in the data set configuration file. A genome build conversion is often accompanied by the loss of some variants when variants in the target build are split into two or more variants or merged into one. This may cause unmatchable or duplicate variants that will subsequently be removed from the data set. A warning will be displayed in the PDF report if a lift–over is requested and more than a pre–defined number of variants are lost due to this step (defaults to 10 %).

Matching of Variant Identifiers with External Databases

Different GWAS platforms often use different variant identifiers for the same variants. In this step, variant identifiers from PLINK’s bim files are matched against a variant database containing identifiers of the HaploType Reference Consortium (HRC) 1.1 reference panel [21], the UK10k cohort [22], the 1000 Genomes Phase 3 Project [23] and the db150 variant database [23]. If a single unique match is found, the respective variant name is retrieved and the original name replaced. In case the variant alleles match the strand complement of the reported alleles in the database, the variant is scheduled for strand alignment in a later step. Some genomic positions are covered by more than one variant, i.e. deletions that span multiple positions and overlap with SNPs, so that it is not clear which variant is to be assigned. In these cases, the name is set to the chr:pos format, e.g. 1:564127. If no match could be found in the database, the original variant name is retained and prefixed with unk_ to indicate that neither name, position, alleles nor strand alignment could be verified, which is often the case if the content is custom chip content. Variants with different names but same position and same alleles are classified as duplicates and one of the duplicates is discarded afterwards.

SNP QC I

During SNP QC I, the prepared data sets, which were pre-processed in parallel in the pre–QC phase, are merged into a single data set and filtered by missingness and Hardy–Weinberg equilibrium (in controls only) across all batches as well as single batches from a particular phenotype and the control group. DeFinetti diagrams [26] are created before and after SNP QC I to verify the quality of variants visually.

Merge Step

If several input data sets have been defined, they are merged into a single PLINK data set. However, the samples will still be analyzed taking into account the previously defined batches.

Missingness

Variants with a high genotyping error rate or which are only present in a subset of the data sets may now be discarded based on variant missingness. Variant missingness is calculated across all batches as well as single batches. The default missing call rate thresholds are 0.1 and 0.02, respectively. If desired, these can be reconfigured by the user.

Tests for Hardy–Weinberg equilibrium

Hardy–Weinberg equilibrium (HWE) tests are performed for controls only in a case–control study (and for all individuals if a quantitative trait is to be considered). HWE tests are performed for all variants from the autosomes (chromosomes 1–22) and the pseudo–autosomal regions (PAR). For women, the complete chromosome X is also tested, if available. Per default, variants deviating from HWE (with an FDR threshold of $10^{-5}$ in controls per default) across the entire batch collection with at most one batch being removed or falling below the FDR threshold in two single control batches are excluded. Batch–wise $p$-values are reported to find variants that fail HWE, among others, in a single control batch (but not in other control batches), in two or more control batches and in the whole collection of controls. Since this version of HWE testing is a computationally intensive task, the data set is divided into small pieces of 1000 variants each, which are then processed in parallel if the surrounding compute environment allows it (e.g. on an HPC cluster). Diagrams are also generated showing how many variants are removed in each specific case; if desired, the user can adjust the threshold parameters using these diagrams.

Sample QC

After SNP QC I, several sample QC measures will be calculated to identify samples showing a high variant missingness rate, an increased or decreased heterozygosity rate, to identify duplicates and related samples and to examine potential population substructure or batch effects by means of principal component analysis (PCA) and Tracy–Widom statistics, with reference samples from HapMap and the 1000 Genomes Project.

Individual Missingness and Heterozygosity

In this step, a list of sample outliers is generated through a series of computations that determine the sample missingness of each data set and the sample–wise heterozygosity rates across
all variants. For the former characteristic, the PLINK tool is run to generate a summary statistic of the genotyping rates per individual sample. Per default, samples that have a missing rate of $\geq 0.02$ are removed. If desired, the user can adjust this threshold. The heterozygosity is determined with a second PLINK call, comparing the expected homozygosity rate with the observed homozygosity rate. Samples with a heterozygosity rate of $\pm 0.5 \times$ standard deviation are classified as outliers. A scatter plot of the proportion called missing (x-axis) against the proportion of SNPs called heterozygote (y-axis) for each individual in the study is generated.

Controlling for population stratification and other confounders
The most common method for identifying (and subsequently removing) individuals with large-scale differences in ancestry is principal component analysis (PCA). To resolve within-Europe relationships (the default) and to test for population stratification, all samples are projected onto principal component (PC) axes generated from four intercontinental HapMap populations as well as 26 reference populations from the 1000 Genomes Project using FlashPCA [28]. The distribution of cases and controls are displayed along the first ten principal components stratified by phenotype. Individual GWAS samples (grey crosses) with values smaller than the median plus/minus three times the interquartile range (median $\pm 1.5 \times$ IQR; samples outside the red square in PCA plot) represent PCA outliers for the first two PCs and are automatically removed (the default; adjustable). All remaining samples (without PCA outliers) are projected onto PC axes generated from 26 reference populations from the 1000 Genomes Project again. Samples are also visualized without reference population samples using FlashPCA, with samples colored by batch code, and the distribution of cases and controls are displayed along the first ten principal components stratified by phenotype. Tracy–Widom statistics [29] are further computed with EIGENSTRAT [30] to evaluate the statistical significance of each PC identified by PCA and to decide which top axes of variation are significant and should be used as covariates in association analyses. For each PCA, a set of independent (MAF $>0.05$) SNPs is calculated, excluding non-autosomes, variants in LD (leaving no pairs with $r^2 \geq 0.2$, within 50 kb windows) or within the extended major histocompatibility complex ($\text{X}$:Chr6:25–34 Mb), and 11 high-LD regions as described by Price et al. [31]. PCA plots with and without A/T and G/C SNPs are created to reveal potential undetected strand problems.

Super population codes from the 1000 Genomes Project are: AFR, African; AMR, Ad Mixed American, EAS, East Asian, EUR, European; SAS, South Asian; CHB, Han Chinese in Beijing, China; JPT, Japanese in Tokyo, Japan; CHS, Southern Han Chinese; CDS, Chinese Dai in Xishuangbanna, China; KHV, Khin in Ho Chi Minh City, Vietnam; CEU, Utah Residents (CEPH) with Northern and Western Ancestry; TSI, Toscani in Italia; FIN, Finnish in Finland; GBR, British in England and Scotland; IBS, Iberian Population in Spain; YRI, Yoruba in Ibadan, Nigeria; LWK, Luhya in Webuye, Kenya; GWD, Gambia; MSL, Mende in Sierra Leone; ESN, Esan in Nigeria; ASW, Americans of African Ancestry in SW USA; ACB, African Caribbeans in Barbados; MXL, Mexican Ancestry from Los Angeles USA; PUR, Puerto Ricans from Puerto Rico; CLM, Colombians from Medellin, Colombia; PEL, Peruvians from Lima, Peru; GIH, Gujarati Indian from Houston, Texas; PJL, Punjabi from Lahore, Pakistan; BEB, Bengali from Bangladesh; STU, Sri Lankan Tamil from the UK; ITU, Indian Telugu from the UK.

SNP QC II
After Sample QC, a few final SNP QC measures are calculated to identify variants where batches within a phenotype and the control group have different allele frequencies even after correction for potential population substructure and variants that show significantly different missingness between cases from the same phenotype and controls.

ANOVA Test
Significantly different allele frequencies of variants across the batches from a particular phenotype or the control group due to batch effects can be detected via analysis of variance (ANOVA) which adjusts for potential population structure from PCA analysis. This test for difference in the mean allele frequencies (after correcting for population stratification) across different batches from a particular disease or the control group. First, allele dosages are regressed against top PCs from PCA and residuals were obtained. P values are calculated from the ANOVA between the residuals of the genotypes and the batches. The null hypothesis is that all groups from a particular disease or from the group are simply random samples of the same population. Variants that have significantly different allele frequencies across the batches (with allowing a single batch to be removed) within phenotypic sets with an FDR threshold of 0.01 (the default) are removed. Robust detection of outlier variants is only possible with the availability of several different batches for the same diagnosis. Hence, this step is only performed if there is at least one defined diagnosis with at least five batches.

Differential Missingness
Given a case/control phenotype, the task is to detect platform/batch differences between case and control genotype data which may lead to false positive signals in association analysis. A Fisher’s exact test on case/control missing call counts at each variant will be performed. Any variant which comes up as...
highly significant under this test ($P \leq 10^{-15}$ per default) should be treated with great caution and will be excluded.

**Final Analysis and Report Generation**

In this stage, the final quality controlled (QCed) output has been generated as no more variants or samples will be removed based on quality metrics. To visualize the changes resulting from the removal of variant and sample outliers, a summary based on several PCA plots is generated.

**Variant Call Format Output**

The QCed GWAS data set is also converted to Variant Call Format (VCF) format so that it can be used as input for genotype imputation and association testing tools. Data sets are converted to the VCF 4.2 format [32] and split to chromosome-wise files as expected by many imputation tools such as the TOPMed Imputation Server [12].

**Automatic Liftover to hg38 (optional)**

The final VCF files are currently based on genome build hg19. Services such as the TOPMed imputation server [12], automatically convert VCF files to hg38, making it difficult to merge post–imputation data with pre–imputation data sets prepared in hg19. Optionally, the conversion to hg38 can be performed automatically by the pipeline using the UCSC liftOver Tools [33].

Due to licensing issues, the UCSC liftover tools [34] are not directly included in our implementation, but can be easily connected to our established UCSC liftover interface with a few steps. The same set of VCF files as for hg19 is generated in hg38 to facilitate the use of downstream imputation and association testing toolchains.

**PDF Report Generation**

Many other pipeline tools that provide GWAS QC include a number of program log files and other logs from which the number and IDs of removed variants and samples must be extracted for a final report. This can be a laborious and error–prone process. In our software, we introduce the automatic generation of a sophisticated PDF report that visualizes all changes and analysis of input data sets in plots and tables within a well–structured, single PDF document. A large collection of figures and tables before and after the QC are arranged side by side to allow easy verification of the results. Using the Nextflow engine, all process executions are further recorded in a single trace file. Our reporting software parses this trace file into an execution tree, where all generated files, such as log files, program output and graphics, can be accessed directly by their task name. A collection of custom parsers then uses the execution tree to create a template in the \LaTeX typesetting language which is finally compiled into a PDF file. A working \LaTeX installation is not required since all necessary software is included in the accompanied Singularity image.

**Association Testing Module**

Association analysis in GWAS/PheWAS is used to identify variants and regions of the genome associated with the phenotype of interest or to examine variants of special interest for their impact on a large collection of phenotypes. The most used software tools for single–variant association analysis are PLINK [5] and SAIGE [13], among others. Logistic/linear regression as implemented in PLINK is the de–facto gold standard for association analysis, with logistic/linear mixed models as implemented in SAIGE now being the new standard for biobank-scale GWAS and PheWAS. The Association Testing Module allows association analysis for binary and quantitative traits with PLINK and SAIGE on both imputed and non–imputed GWAS data sets on different genome builds, both with and without covariates, including chromosome X analysis, combined in a single–command. For quantitative traits, the user has to set the parameter \texttt{trait} to "quantitative" instead of "binary" (the default). In this case, PLINK as well as SAIGE will then automatically perform an association analysis for a quantitative trait. All processes, including conversion steps, are largely processed in parallel, so that each GWAS association analysis for each phenotype can be accomplished by a single run (see Figure 3).

**Data Preparation**

No further genotype data preparation step is required if the input data set, a set of chromosome–wise VCF files (optionally including chromosome X), is obtained either from the Quality Control Module or from genotype imputation programs such as the TOPMed Imputation Server and contains dosage and/or genotype information. VCF files must have a DS tag for dosage data or a GT tag for genotyped data. If both are present, DS is chosen.

**Sample Selection, Covariate Preparation and Null Model Building for SAIGE**

To select a desired subset of samples for association analysis, an optional PLINK FAM file to update sex information and case–control status can be specified, and an optional covariate file if covariate adjustment is desired. In this case only the joint set of samples from the sample list of the VCF and the optional FAM file (and the optional covariate file) is used for model building (in SAIGE) and association analysis; no further modification of the VCF input files is necessary. For model building in SAIGE using saddle point approximation (SPA) [13] in order to calibrate unbalanced case–control ratios, the null logistic/linear mixed model is fitted based on a set of independent variants (see section Controlling for population and other confounders, Methods), with all independent variants also having an imputation quality score of $\hat{r}^2 \geq 0.8$ (threshold configurable). With the same set of independent variants PCA analysis is performed using FlashPCA2 [28] to calculate top PCs (default n=10) to ad-
just for potential population stratification.

**Plink Association Testing**

PLINK’s logistic/linear regression association module requires a custom text-based file with genotype dosage information (genotype probability format) which is generated on-the-fly from imputed or non-imputed VCF input files. Genotype dosage-based association analysis is conducted with the top PCs (default n=10) from PCA and user-defined optional covariates from the covariate file. For chromosome X association analysis, haplotypic allele calls outside the PAR regions in males are converted to homozygous calls by doubling the haplotypic allele (assuming inactivation of large parts of one of the two female X chromosomes [35]), and sex is used as a covariate for association testing of the non-PAR regions on chromosome X.

**SAIGE Association Testing**

One of the main features of SAIGE is its ability to perform single-marker association testing for extremely unbalanced case-control ratios and rare variants [13]. While SAIGE’s computational load is comparable to logistic or linear regression as performed by PLINK, SAIGE can directly read compressed VCF files typically generated from genotype imputation programs, including imputed dosage data and genotype-only calls, and further supports the memory efficient BGEN format as used in the UK Biobank imputed data releases. SAIGE allows much finer control on parallelization by allowing the association analysis on specified genomic parts of a VCF/BGEN without the need to cut the file into smaller chunks. This technique does not only save runtime that would otherwise be consumed by splitting up VCF files but also large amounts of temporary file storage and I/O load. As no extra computational load is required for splitting of chromosomal files, the scalability of how many jobs can be executed in parallel is only limited by the available computational resources, and the overhead is very low with respect to the chunk size of the genomic parts. Using only a sub-sample of all samples from VCF input files is not supported by SAIGE yet. For this reason, we have written a routine that automatically extracts a sub-sample from the VCF files based on a user-defined PLINK FAM file before starting the analysis with SAIGE. Association analysis with SAIGE (mixed effect model + covariates) is conducted using the top PCs (default n=10) from PCA and user-defined optional covariates from the covariate file. The same chromosome X analysis method used by PLINK is performed in SAIGE.

**Compatibility with UK Biobank Data**

The UKB’s latest GWAS data release contains, in addition to unimputed genotype data in PLINK format, pre-imputed data in the BGEN format instead of the much more common VCF format. However, most computational procedures of the Association Testing Module, especially the conversion routines that convert VCF files to PLINK’s genotype dosage format, heavily rely on the VCF format specification. BGEN, while much more efficient than VCF, is a binary format that is not designed to be processed by scripts, and as such, cannot directly be processed into PLINK’s genotype dosage format. Since the more common VCF format is a generic format used by biobanks and genotype imputation programs (TOPMed and Sanger Imputation server [12], among others) worldwide, we chose to drop BGEN support and therefore, in addition to SAIGE, can still keep PLINK as a de-facto gold standard tool for regression-based association analysis in our association testing workflow.

However, we included a set of scripts to perform SAIGE association testing on UKB BGEN files. This “pipeline” is a feature-reduced version that only performs the absolutely necessary steps to run SAIGE on the input data but uses the same fine-grained parallelization techniques as used in the SAIGE/Plink VCF-based pipeline. External covariates, such as principal components, sex and age, can be extracted from UKB sample information databases and supplied to the association testing process.

**Implementation Details**

We use the free and open-source Nextflow [14] pipeline software for easy parallelization of processes, HPC-based development and pipeline control. Singularity, a container format and software focused on scientific computation and reproducibility, is used to provide a single-command experience without the need to install an entire software execution environment of different software packages.

**Nextflow**

While both the QC Control Module and the Association Testing Module are on the same level of runtime performance as comparable GWAS pipelines that analyse a smaller number of samples and variants (compare Tables 2 and 3), the real challenge of existing GWAS pipelines becomes apparent with larger data sets. For example, many QC and association testing tools use the duplicate/relatedness detection feature (see Methods) to remove related individuals or duplicate samples from the data set. As every sample is tested against every other sample, the total runtime grows quadratic in relation to the number of samples, i.e. the time complexity is \(O(n^2)\).

Nextflow, a tool specifically designed to run bioinformatics applications, therefore provides a suitable environment to facilitate the parallelism of processes in all workflows. It extends the Unix pipes model with a fluent data description language (DSL), allowing you to handle complex stream interactions easily. In our case, we use a dynamic job creation approach to automatically determine the number of parallel jobs required to perform an operation within a reasonable time frame. For example, in the duplicate/relatedness detection step, we use both PLINK’s “parallel” mode and Nextflow’s job splitting feature to achieve fast processing of large input files. The same mechanism is also used for other computationally intensive operations, such as our “leave-one-batch-out Hardy-Weinberg test” or the SAIGE association testing procedure. Nextflow takes care of the creation, workspace separation and merging of all jobs, avoids user-defined error-prone synchronization of all jobs, and the merging of results from complex analysis pipelines. All the intermediate results produced during the pipeline execution are automatically tracked.

Another major advantage of Nextflow is the already integrated handling and agnostic use of HPC cluster resources (i.e. it is able to work under different environments). With Nextflow, it is possible to write parallel and concurrent pipeline workflows without having knowledge (and thus knowing all dependencies) of a specific cluster environment, such as SGE, SLURM or PBS/TORQUE. This makes it possible to provide pipelines for a wide range of HPC and non-HPC users. The deep integration of different storage solutions like ECS object storage or S3 buckets and container tools such as Docker and Singularity make Nextflow a first choice for a portable, widely available platform, with the only requirement of a working Java installation.
Singularity

Our general Nextflow script code base is hosted on GitHub. However, the included pipeline scripts make heavy use of external tools, such as the well-known PLINK software [5] for GWAS data. Each workflow is usually created with specific versions of tools in mind, it is often the case that developers need to install multiple versions of the same software and must take special care in managing them. Conda [36] and Lmod [37] assist in setting up the respective software environments, but the required tools must still be installed manually by the user.

In our case we decided to prepare the execution environment, including an entire operating system, in a publicly-hosted container image, including all specific program versions. This not only eliminates the burden and error-proneness of manually creating a new environment, but also ensures that each user executes exactly the same program versions and therefore can reproduce any result with the same input, without any problems with deployment on different operating systems, user permissions or configuration management.

Another advantage of using an entire file system is not only the possibility to add programs as desired, but to include any file. We use these capabilities to provide ready-to-use Hapmap2 [38] and 1000 Genomes [25] data sets that serve as reference samples, for example, for principal component analysis (PCA), as well as a comprehensive but slim database of NCBI Reference SNP ID numbers [39].

Results

Benchmark Settings

In Table 1 we compare the features of the QC Module of BIGwas (for details see Methods) with the features of the QC module from the H3Agwas Pipeline Version 3, a portable and reproducible GWAS QC and association test software workflow developed by the pan-African community project H3ABioNet (see Introduction). The H3A QC module was developed to perform a robust and efficient QC workflow for over 30,000 DNA samples genotyped using a custom designed African genotyping array as part of H3ABioNet [11]. Since many similar features were implemented in both software pipelines (although there is no explicit support for multiple GWAS data sets (i.e. batches) in H3A QC), the runtime performance can be compared very well in a benchmark. We have not made a feature comparison for the association pipelines of BIGwas (for details see Methods) and H3A, since the authors of the H3A Association Pipeline have made it clear that the purpose of the H3A Association Pipeline is to perform a very basic logistic/linear regression association analysis with PLINK (https://github.com/h3abionet/h3agwas/). The use of linear mixed models for association testing is further supported, but only for genotypes in PLINK format (and not imputed data) and without support for pre- or post-processing of input and output files. For this reason, we only compared the runtime of logistic/linear regression and mixed model testing with BIGwas (PLINK and SAIGE) versus logistic/linear regression with H3A (PLINK).

We conducted benchmark tests on our HPC system with 68 computational nodes and 1,336 CPU cores in total, which enables highly parallel computing. However, since in a multi-user HPC system not all resources are usually available to a single user, and in order to demonstrate that BIGwas uses resources economically, for benchmarking we allowed submitting a maximum of 150 jobs in parallel (corresponds to 7 compute nodes; configurable by the user) for testing both the QC Module and the Association Testing Module of BIGwas. For benchmarking H3A, we made no restrictions on HPC resources (thus allowing 1,336 jobs in parallel) to test the maximum performance of H3A. For each benchmark data set consisting of a different number of GWAS samples and genetic variants (Table 2 and Table 3), we used Nextflow’s built-in reporting function to determine the elapsed wall-clock time. Each run-time analysis was performed three times on an empty HPC cluster to account for potential fluctuations in HPC cluster performance.

Table 1. Feature comparison between the QC Module of BIGwas (for details see Methods) and the H3A QC module of the H3Agwas Pipeline Version 3 [11]. The QC Module of BIGwas performs, among other things, an automatic upscaling of job parallelization with respect to the number of samples and variants and allows processing an unlimited number of input GWAS data sets (i.e. batches) with a single command.

| Feature | BIGwas QC | H3A QC |
|---------|-----------|--------|
| Duplicate Removal | ✓ | ✓ |
| RsID dbSNP Assignment | ✓ | ✗ |
| Hardy–Weinberg Equil. (HWE) Test | ✓ | ✓ |
| HWE Test (batch-wise leave-one-out) | ✓ | ✗ |
| IBD/Relatedness Check | ✓ | ✓ |
| Missings Check | ✓ | ✓ |
| Heterozygosity Check | ✓ | ✓ |
| PCA Outlier Detection | ✓ | ✗ |
| Low–MAF Removal | ✗ | ✓ |
| Pseudautosomal Region (PAR) Check | ✓ | ✗ |
| VCF Output | ✓ | ✓ |
| Genome Build Liftover | optional | ✗ |
| Dynamic Parallelization of Jobs | ✓ | ✗ |
| PDF Report | ✓ | ✓ |

1 Principal component analysis (PCA) is calculated but ancestry outliers are not determined
2 Parallelization effort is manually configurable but does not automatically scale

Table 2. Runtime Metrics of the QC Module of BIGwas compared to the QC module of the H3Agwas Pipeline Version 3 [11] for several different-sized GWAS data sets. A GWAS input data set with 976,584 samples and 803,113 genetic variants (i.e. twice the size of the current UK Biobank GWAS data set) can be quality-controlled in less than 7 days with one command of the BIGwas software, whereby only 150 jobs (configurable; equivalent to about 7 compute nodes on our HPC cluster system) are used in parallel. (*) indicates that the runtime of H3A has exceeded the maximum time allocation of 10 days for the HPC system despite the possibility to use a maximum of 1,336 jobs in parallel.

| Samples | Variants | BIGwas Runtime | H3A Runtime |
|---------|----------|----------------|-------------|
| 200     | 500      | 6 min          | 4 min       |
| 500     | 1,000    | 6 min          | 7 min       |
| 1,000   | 5,000    | 6 min          | 9 min       |
| 5,000   | 50,000   | 8 min          | 18 min      |
| 5,000   | 250,000  | 15 min         | 35 min      |
| 10,000  | 250,000  | 15 min         | 57 min      |
| 20,554  | 700,078  | 2h 15 min      | 8h 57 min   |
| 488,292 | 231,151  | 3d 1h          | *           |
| 488,292 | 803,113  | 4d 22h         | *           |
| 976,584 | 803,113  | 6 d 12h        | *           |
Benchmark Results

Table 2 shows runtime benchmarks of the QC Module of BIGwas (for details see Methods) compared to the QC module of the H3Agwas Pipeline Version 3. For comparison with H3A runtimes, we set configurable parameters, such as missingness thresholds, to similar values where possible and used the recommended commands for the respective benchmark data sets. Data sets in the upper part were subsampled from in-house GWAS data sets genotyped using Illumina’s Global Screening (GSA) Array v1.0. In the lower part, unimputed Axiom Array data of 488,292 individuals of European ancestry from UKB was used (see Acknowledgements). The data set of 976,584 samples and 803,113 SNPs represents an artificially created GWAS data set (from the original COVID-19 UK Biobank unimputed GWAS data release, September 2020, with 1,485 COVID-19 cases and 486,807 controls), which was created by duplicating the number of samples from the UK Biobank data set.

The theoretical algorithmic runtime complexity of the QC is quadratic with respect to the number of samples and linear with the number of variants $O(n^2 + m)$, since in identity-by-descent (IBD) analysis each sample is checked against every other sample both in the H3A as well as BIGwas. The IBD calculation becomes the most time-consuming part from 5,000 samples upwards in H3A QC and leads to a preliminary termination of H3A QC for the UK data set (Table 2). For 5,000 GWAS samples upwards, our implementation begins with a dynamic parallelization process (see Methods), i.e. scheduling (at maximum 150) jobs (configurable) to distribute the workload across multiple compute nodes in order to reduce the runtime (not shown) with respect to the number of samples. The number of additional jobs scales with the number of samples to keep the runtime for IBD calculation low (see processes marked with $\times n$ in Figure 2), thus the runtime of BIGwas increases much slower than that of H3A.

Algorithmic complexity in other time-consuming applications such as Hardy–Weinberg and ANOVA testing is generally linear in the number of variants but we found it useful to also scale the number of parallel jobs or threads automatically with the number of samples (also marked with $\times n$ in Figure 2). Our resource-efficient intra- and inter-node software implementation (see Methods) shows that BIGwas did not exceed the maximum number of 150 parallel jobs allowed by us in our benchmark, and that BIGwas is able to analyze arbitrarily large GWAS input data sets with one million GWAS samples. Despite the possibility to use a maximum of 1,336 parallel jobs, H3A exceeded a runtime of 10 days for the UK data set with 488,292 samples and a reduced set of 231,151 (linkage disequilibrium pruned) variants (Table 2). Results of BIGwas are summarized in a pdf report (see Methods) and output files (on either genotype build hg19 and/or hg38; configurable) can be used directly for association testing and/or as an input for genotype imputation services.

Table 3 shows runtime benchmark results of the Association Testing Module of BIGwas (for details see Methods) compared to the association module of H3A. Data sets in the upper part were subsampled from in-house GWAS data sets genotyped using Illumina’s Global Screening (GSA) Array v1.0. In the lower part, pre-imputed GSA data of 5,480 individuals as well as pre-imputed Axiom Array data of 487,409 individuals of European ancestry from UKB was used (see Acknowledgements). The data set of 974,818 samples and 92,775,302 SNPs represents an artificially created imputed GWAS data set (from the original COVID-19 UK Biobank imputed GWAS data release, September 2020, with 636 cases and 486,773 controls), which was created by duplicating the samples of the UKB data set. Association analysis included top 10 principal components from PCA, sex, age, sex*age and age*age as covariates.

The theoretical runtime of BIGwas increases linear with the number of variants $O(m)$. The runtimes shown in Table 3 were achieved with a maximum number of parallel jobs of 150 for BIGwas, and with no restrictions on the number of parallel jobs for H3A (at maximum 1,336 jobs in parallel). By default, our pipeline module schedules one job for 5,000 genetic variants (marked with $\times m$ in Figure 3) for the limited number of maximum 150 parallel jobs (configurable). Our resource-efficient intra- and inter-node software implementation (see Methods) enables BIGwas to analyze arbitrarily large GWAS input data sets with millions of GWAS samples and millions of imputed genetic variants. Although SAIGE and PLINK association testing in BIGwas is performed in parallel in one run of the Association Testing Module, the runtime of BIGwas is only slightly longer than that of H3A, which performs association testing with PLINK only, without pre- and post-processing of input and output files. Some new features have been implemented in the Association Testing Module of BIGwas, which are not covered by the original implementations of SAIGE and PLINK (see Methods): Before association tests are performed, an on-the-fly sample filtering is conducted to exclude certain samples (e.g. based on a sample exclusion list; see Methods) or variants (e.g. based on the imputation score threshold $r^2$; see Methods) from PLINK or VCF input data sets, without the need to manually remove them first, which is usually a labor-intensive step for the user. In addition, an automated principal component analysis (PCA; configurable) is integrated into SAIGE mixed-model association analysis, so that model fitting analysis and association testing (along with principal components from PCA and/or user provided covariates) are performed in SAIGE in a single step; association testing for chromosome X is also automated for PLINK and SAIGE (see Methods). Association summary statistics output files as well as best-guess genotype calls in PLINK format are produced for different genome builds (hg19 and/or hg38; configurable).

Table 3. Runtime Metrics of the Association Testing Module of BIGwas (regression analysis with PLINK and mixed model analysis with SAIGE) compared to association testing module of the H3Agwas Pipeline Version 3 (regression analysis with PLINK) [11] for several different-sized GWAS data sets. An imputed genome-wide input data set with almost one million samples and more than 92 million genetic variants (i.e. twice the size of the imputed UKB GWAS data set) can be tested for association within 10 days with one command of the BIGwas software, whereby only 150 jobs (configurable; equivalent to about 7 compute nodes on our HPC cluster system) are used in parallel. (*) indicates that H3A uses PLINK genotype files as input (but not imputed allele dosages).

| Samples | Variants | BIGwas Runtime | H3A Runtime |
|---------|----------|----------------|-------------|
| 10,000  | 250,000  | 21 min         | 8 min       |
| 10,000  | 700,078  | 54 min         | 16 min      |
| 20,554  | 700,078  | 1h 33 min      | 1h 23 min   |
| 5,480   | 81,708,012| 14 h 26 min    | *           |
| 487,409 | 92,775,302| 4.18 h         | *           |
| 974,818 | 92,775,302| 9.14 h         | *           |

Discussion

We have introduced BIGwas, a portable (i.e. Singularity container-based), fully automated GWAS quality control (QC) and association pipeline in Nextflow that can be executed with a single command without manual installation by the user. BIGwas differs from existing GWAS data processing packages in R in that it allows standardized, reproducible, and high-throughput processing of GWAS data sets with a single...
command, whereas existing GWAS R packages allow for more flexible processing, for example, with respect to the order of processing steps. The current UKB GWAS release with 487,409 GWAS samples can be analysed with BIGwas within 8 days (4 days for QC and 4 days for association testing) on a small HPC with 7 compute nodes. For a GWAS data set with up to one million individuals BIGwas needs on average 16 days on a HPC with 7 compute nodes. Using our dynamic parallelization approach, shorter runtimes can be achieved with a larger HPC. For quality control (QC) the theoretical runtime increases quadratically with the number of samples and linear with the number of variants $O(n^2 + m)$. For association testing the runtime increases linear with the number of variants $O(m)$. As sample sizes in large GWAS/PheWAS biobank studies, for example, for COVID-19 research (https://www.covid19hg.org/), have grown to millions of GWAS samples and GWAS data sets of this size are likely to be used more frequently in the future, the efficiency and ease of execution of BIGwas with a single command will become an important criterion for conducting GWAS/PheWAS studies.

To enable a reproducible analysis of one million GWAS samples on any common computer platform, we have combined various code optimizations of our existing GWAS software with Nextflow workflow and Singularity container technology. Our Nextflow pipelines are inherently parallel and can be transparently scaled up or down, and they provide an abstraction layer between the logic of the pipeline and the execution layer (i.e. they can be executed on multiple platforms without changing the source code). With the Singularity container technology, we have packed all our Nextflow workflows, software and libraries, as well as publicly-available reference data into a single Singularity container that runs on any standard computer platform without manual download or installation by the user.

BIGwas implements a complete GWAS analysis protocol which was developed for large-scale GWAS studies of the Inflammatory Bowel Disease Genetics Consortium (IIBDGC) and the Severe Covid-19 GWAS Group [17, 18, 19]. BIGwas is not yet suitable for GWAS analysis of family data or GWAS analysis across populations of worldwide ethnicities. To enable these analyses, extensions to our QC and the association testing pipelines or the use of alternative statistical methods may be required. In the future, we aim to test the usability of further genome-wide analysis approaches, such as gene–gene interaction tests, for use with Nextflow and Singularity technology.

### Availability of source code and requirements

- **Project name:** IKMB GWAS QC and Association Testing Pipelines
- **Project home page:** [http://github.com/ikmb/gwas-qc](http://github.com/ikmb/gwas-qc) and [http://github.com/ikmb/gwas-assoc](http://github.com/ikmb/gwas-assoc)
- **Operating system:** Linux
- **Programming language:** Nextflow, Bash, Python, Perl, R, Rust
- **Other requirements:** Java 8 or higher, Singularity 3.4 or higher
- **License:** MIT

This project makes use of several open source tools, namely Plink 1.9 [40], Plink 2 [5], bcftools 1.10 from the HTSlib project [41], FlashPCA 2.0 [28], EIGENSTRAT 4.2 [30] and SAIGE 0.42.1 [13]. For the availability of source code, we refer to the respective publications. Several more native binaries were created and included for performance reasons. Their source codes are available as follows:

- **chipmatch**, a tool to match Plink’s BIM files against the Will Rayner’s strand database [22]: [https://github.com/ikmb/chipmatch](https://github.com/ikmb/chipmatch)
- **chipmatch**, a high-performance heat map plotting tool for almost arbitrary, very large data sets: [https://github.com/ikmb/genericplotter](https://github.com/ikmb/genericplotter)

**UK Biobank (UKB) COVID-19 GWAS data release (September 2020):** [https://biobank.ndph.ox.ac.uk/ukb/exinfo.cgi?src=COVID19](https://biobank.ndph.ox.ac.uk/ukb/exinfo.cgi?src=COVID19)

**Availability of supporting data and materials**

An example data set is included in the software packages, see project home page. The example data set is a subset of the 1000 Genomes Project, using 2504 samples with randomized sex and phenotype, and 50,000 randomly-selected variants.

### Declarations

**Ethics approval and consent to participate**

All participants provided written informed consent, and the study was approved by the ethics boards of the participating institutions (see Acknowledgements) in agreement with the Declaration of Helsinki principles.

**Consent for publication**

Not applicable.

**Competing Interests**

The authors declare that they have no competing interests.

**Author’s Contributions**

**Jan Christian Kässens:** Conceptualization, Methodology, Software, Benchmarks, Writing **Lars Wienbrandt:** Software, Benchmarks, Writing, Visualization **David Ellinghaus:** Conceptualization, Methodology, Software, Formal analysis, Data curation, Supervision, Writing

### Acknowledgements

The authors would like to thank Frauke Degenhardt for sharing R scripts for visualisation of results from principal component analysis and Florian Uellendahl–Werth and Mareike Wendorff for helpful discussions on R code optimization. Benchmarks and scalability tests have been conducted using the UK Biobank Resource (Project #61199, “Biobank–Scale Configurable Quality Control and Association Testing for GWAS”). This work was supported by the German Federal Ministry of Education and Research (BMBF) within the framework of the Computational Life Sciences funding concept (ComplLS grant 031L0165) and iTREAT (grant 01ZX1902A). The study received infrastructure support from the DFG Cluster of Excellence 2167 “Precision Medicine in Chronic Inflammation (PMI)” (DFG Grant: “EXC2167”), the DFG research unit “miTarget” (Projectnumber 426660215) and the PopGen Biobank (P2N 2016–101).
References

1. Evangelou E, Warren HR, Mosen-Ansorena D, Mifsud B, Pazoki R, Gao H, et al. Genetic analysis of over 1 million people identifies 535 new loci associated with blood pressure traits. Nature Genetics 2018 Sep;50(10):1422–1425. https://doi.org/10.1038/s41588-018-0205-x.

2. Sudlow CJ, Gallagher J, Allen N, Beral V, Burton P, Danesh J, et al. UK Biobank: An Open Access Resource for Identifying the Causes of a Wide Range of Complex Diseases of Middle and Old Age. PLOS Medicine 2015 Mar;12(3):e1001779. https://doi.org/10.1371/journal.pmed.1001779.

3. Klarin D, Damrauer SM, Cho K, Sun YV, Teslovich TM, Honerlaw J, et al. Genetics of blood lipids among ~300,000 multi–ethnic participants of the Million Veteran Program. Nature Genetics 2018 Oct;50(11):1514–1523. https://doi.org/10.1038/s41588-018-0222-9.

4. Talieu SAG, VandeHaar P, Boughton AP, Welch RP, Talieu D, Schmidt EM, et al. Exploring and visualizing large-scale genetic associations by using PheWeb. Nature Genetics 2020 Jun;52(6):550–552. https://doi.org/10.1038/s41588-020-0162-5.

5. Chang CC, Chow CC, Tellier LC, Vattikuti S, Purcell SM, Lee JJ. Second-generation PLINK: rising to the challenge of larger and richer datasets. GigaScience 2015 Feb;4(1). https://doi.org/10.1186/s13742-015-0047-8.

6. Gogarten SM, Bhangale T, Conomos MP, Laurie CA, McHugh CP, Painter I, et al. GWASTools: an R/Bioconductor package for quality control and analysis of genome-wide association studies. Bioinformatics 2012 Oct;28(24):3329–3331. https://doi.org/10.1093/bioinformatics/bts610.

7. van der Most PJ, Vaez A, Prins BP, Munoz ML, Snieder H, Alizadeh BZ, et al. QCGWAS: A flexible R package for automated quality control of genome–wide association results. Bioinformatics 2014 Jan;30(8):1185–1186. https://doi.org/10.1093/bioinformatics/btu745.

8. Perreault LPL, Legault MA, Asselin G, Dubé MP. genie: an automated genome-wide imputation pipeline with automatic reporting and statistical tools. Bioinformatics 2016 Aug;32(17):2582–2584. https://doi.org/10.1093/bioinformatics/btw487.

9. Lam M, Awasthi S, Watson HJ, Goldstein J, Panagiotou G, Trubetskoy V, et al. RICOPILLI: Rapid Imputation for COSORTias PipeLine. Bioinformatics 2019 Aug;35(16):2567–2572. https://doi.org/10.1093/bioinformatics/btz633.

10. Ellinghaus D, Jostins L, Spain SL, Cortes A, Bethune J, Han B, et al. Analysis of five chronic inflammatory diseases identifies 27 new associations and highlights disease-specific patterns at shared loci. Nature Genetics 2016 Mar;48(5):510–518. https://doi.org/10.1038/ng.3528.

11. Zhou W, Fang M, Jostins L, Mirkov MU, Boucher G, Anderson CA, et al. Fine-mapping inflammatory bowel disease loci to single-variant resolution. Nature 2017 Jun;547(7662):173–178. https://doi.org/10.1038/nature22969.

12. Rayner NW, McCarthy MI. Development and Use Of A Pipeline to Generate Strand and Position Information for Common Genotyping Chips; 2011, poster presented at the ASHG 2011, Montreal, Canada.

13. McCarthy S, Das S, Kretschmar W, Delaneau O, Wood AR, Teumer A. A reference panel of 64,976 haplotypes for genotype imputation. Nature Genetics 2016 Aug;48(10):1279–1283. https://doi.org/10.1038/ng.3528.

14. Kurtzner GM, Sochat V, Bauer MW. Singularity: Scientific containers for mobility of compute. PLOS ONE 2017 May;12(5):e0177459. https://doi.org/10.1371/journal.pone.0177459.

15. Anderson CA, Pettersson FH, Clarke GM, Cardon LR, Morris AP, Zondervan KT. Data quality control in genetic case-control association studies. Nature Protocols 2010 Aug;5(9):1564–1573. https://doi.org/10.1038/nprot.2010.116.

16. Anderson CA, Pettersson FH, Clarke GM, Cardon LR, Morris AP, Zondervan KT. Data quality control in genetic case-control association studies. Nature Protocols 2010 Aug;5(9):1564–1573. https://doi.org/10.1038/nprot.2010.116.

17. Ellinghaus D, Jostins L, Spain SL, Cortes A, Bethune J, Han B, et al. Analysis of five chronic inflammatory diseases identifies 27 new associations and highlights diseasespecific patterns at shared loci. Nature Genetics 2016 Mar;48(5):510–518. https://doi.org/10.1038/ng.3528.

18. Huang H, Fang M, Jostins L, Mirkov MU, Boucher G, Anderson CA, et al. Fine-mapping inflammatory bowel disease loci to single-variant resolution. Nature 2017 Jun;547(7662):173–178. https://doi.org/10.1038/nature22969.
12 | GigaScience, 2017, Vol. 00, No. 0

ture Genetics 2006 Jul;38(8):904–909. https://doi.org/10.1038/ng1847.

31. Price AL, Weale ME, Patterson N, Myers SR, Need AC, Shianna KV, et al. Long-Range LD Can Confound Genome Scans in Admixed Populations. The American Journal of Human Genetics 2008 Jul;83(1):132–135. https://doi.org/10.1016/j.ajhg.2008.06.005.

32. Danecek P, Auton A, Abecasis G, Albers CA, Banks E, DePristo MA, et al. The variant call format and VCFtools. Bioinformatics 2011 Jun;27(15):2156–2158. https://doi.org/10.1093/bioinformatics/btr330.

33. Kuhn RM, Haussler D, Kent WJ. The UCSC genome browser and associated tools. Briefings in Bioinformatics 2012 Aug;14(2):144–161. https://doi.org/10.1093/bib/bbs038.

34. Haeussler M, Zweig AS, Tyner C, Speir ML, Rosenbloom KR, Raney BJ, et al. The UCSC Genome Browser database: 2019 update. Nucleic Acids Research 2018 Nov;47(D1):D853–D858. https://doi.org/10.1093/nar/gky1096.

35. König IR, Loley C, Erdmann J, Ziegler A. How to Include Chromosome X in Your Genome-Wide Association Study. Genetic Epidemiology 2014 Jan;38(2):97–103. https://doi.org/10.1002/gepi.21782.

36. Continuum Analytics, Inc. Package, dependency and environment management for any language—Python, R, Ruby, Lua, Scala, Java, JavaScript, C/ C++, FORTRAN, and more.; 2017 (accessed October 16, 2020), https://docs.conda.io/en/latest/.

37. McLay R, Schulz KW, Barth WL, Minyard T. Best practices for the deployment and management of production HPC clusters. In: State of the Practice Reports on – SC ’11 ACM Press; 2011. p. 1–11. https://doi.org/10.1145/2063348.2063360.

38. International HapMap Consortium, Frazer KA, Ballinger DG, Cox DR. A second generation human haplotype map of over 3.1 million SNPs. Nature 2007 Oct;449(7164):851–861. https://doi.org/10.1038/nature06258.

39. Kitts A, Sherry S. The Single Nucleotide Polymorphism Database (dbSNP) of Nucleotide Sequence Variation. In: McEntyre J, Ostell J, editors. The NCBI Handbook Bethesda (MD), US: National Center for Biotechnology Information; 2011.p. 1–34. Available from https://www.ncbi.nlm.nih.gov/books/NBK21088/.

40. Purcell S, Neale B, Todd–Brown K, Thomas L, Ferreira MAR, Bender D, et al. PLINK: A Tool Set for Whole-Genome Association and Population-Based Linkage Analyses. The American Journal of Human Genetics 2007 Sep;81(3):559–575. https://doi.org/10.1086/519796.

41. Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, et al. The Sequence Alignment/Map format and SAMtools. Bioinformatics 2009 Jun;25(16):2078–2079. https://doi.org/10.1093/bioinformatics/btp352.
Revised manuscript; Submission ID GIGA-D-20-00360: “BIGwas - Single-command Quality Control and Association Testing for biobank-scale GWAS/PheWAS data”

Dear Dr. Edmunds,

please find enclosed our revised manuscript and point-by-point response that addresses the reviewers’ concerns.

We have detailed all additional software implementations and text changes in the point-by-point response. On our official GitHub tutorial websites https://github.com/ikmb/gwas-qc/ and https://github.com/ikmb/gwas-assoc, we have updated our software and documented the new software features for the review process.

With the help of our colleague Dr. Lars Wienbrandt, we have extended the quality control module for the analysis of quantitative features. Therefore, Lars Wienbrandt is now also co-author of the manuscript. In addition, we have implemented a number of improvements, including improved memory management resulting in a 20% reduction in input/output load when creating temporary files with Nextflow.

We hope that we have been able to address all concerns expressed and that you continue to share our enthusiasm for our software. We are looking forward to your decision and are at your disposal for further questions.

Yours faithfully,

Jan Christian Kässens, Ph.D.  
Lars Wienbrandt, Ph.D.  
David Ellinghaus, Ph.D.