1 **The variant call format provides efficient and robust storage of GWAS summary statistics**

2

3 Matthew Lyon (0000-0002-2500-1013)[1,2]*, Shea J Andrews (0000-0002-1921-9470)[3]*, Ben

4 Elsworth (0000-0001-7328-4233)[2], Tom R Gaunt (0000-0003-0924-3247)[1,2], Gibran Hemani

5 (0000-0003-0920-1055)[2]¥, Edoardo Marcora (0000-0002-3829-4927)[3]¥

6

7 1. National Institute for Health Research Bristol Biomedical Research Centre, University of

8   Bristol, Bristol, UK

9 2. Medical Research Council (MRC) Integrative Epidemiology Unit (IEU), Bristol Medical

10   School (Population Health Sciences), University of Bristol, Bristol, UK

11 3. Ronald M. Loeb Center for Alzheimer's disease, Department of Neuroscience, Icahn

12   School of Medicine at Mount Sinai, New York, NY, USA

13

14 * These authors contributed equally to this work

15 ¥ These authors contributed equally to this work

Genome-wide association study (GWAS) summary statistics are a fundamental resource for a variety of research applications [1–6]. Yet despite their widespread utility, no common storage format has been widely adopted, hindering tool development and data sharing, analysis and integration. Existing tabular formats [7,8] often ambiguously or incompletely store information about genetic variants and their associations, and also lack essential metadata increasing the possibility of errors in data interpretation and post-GWAS analyses. Additionally, data in these formats are typically not indexed, requiring the whole file to be read which is computationally inefficient. To address these issues, we propose an adaptation of the variant call format [9] (GWAS-VCF) and have produced a suite of open-source tools for using this format in downstream analyses. Simulation studies determine GWAS-VCF is 9-46x faster than tabular alternatives when extracting variant(s) by genomic position. Our results demonstrate the GWAS-VCF provides a robust and performant solution for sharing, analysis and integration of GWAS data. We provide open access to over 10,000 complete GWAS summary datasets converted to this format (available from: https://gwas.mrcieu.ac.uk).

31   **Main**

32

33   The GWAS is a powerful tool for identifying genetic loci associated with any trait, including

34   diseases and clinical biomarkers, as well as non-clinical and molecular phenotypes such as

35   height and gene expression [3] (eQTLs). Sharing of GWAS results as summary statistics (i.e.

36   variant, effect size, standard error, p-value etc.) has enabled a range of important secondary

37   research applications including: causal gene and functional variant prioritisation [1], causal

38   cell/tissue type nomination [2], pathway analysis [3], causal inference (Mendelian

39   randomization; MR) [4], risk prediction [3], genetic correlation [5] and heritability estimation [6].

40   However, the utility of GWAS summary statistics is hampered by the absence of a

41   universally adopted storage format and associated tools.

42

43   Historic lack of a common standard has resulted in GWAS analysis tools outputting summary

44   statistics in different tabular formats (e.g. plink [10], GCTA [11], BOLT-LMM [12], GEMMA [13], Matrix

45   eQTL [14] and meta-analysis tools e.g. METAL [15]). As a consequence, various processing issues

46   are typically encountered during secondary analysis. First, there is often inconsistency and

47   ambiguity of which allele relates to the effect size estimate (the "effect" allele). Confusion

48   over the effect allele can have disastrous consequences on the interpretation of GWAS

49   findings and the validity of post-GWAS analyses. For example MR studies may provide

50   causal estimates with incorrect effect directionality [16]. Likewise, prediction models based on

51   polygenic risk scores might predict disease wrongly or suffer reduced power if some of the

52   effect directionalities are incorrect. Second, the schema (i.e. which columns/fields are

53   included and how they are named) of these tabular formats varies greatly. Absent fields can

54   limit analyses and although approaches exist to estimate the values of some of these

55 missing columns (e.g. standard error from P value) imprecision is introduced reducing

56 subsequent test power. Varying field names are easily addressed in principle, but the

57 process can be cumbersome and error prone. Third, data are frequently distributed with no

58 or insufficient metadata describing the study, trait(s), and variants (e.g., trait measurement

59 units, variant id/annotation sources, etc.) which can lead to errors, impede integration of

60 results from different studies and hamper reproducibility. Fourth, querying unindexed text

61 files is slow and memory inefficient, making some potential applications computationally

62 infeasible (e.g. systematic hypothesis-free analyses).

63

64 Some proposals for a standard tabular format have been made. The EBI-NHGRI GWAS

65 catalog (www.ebi.ac.uk/gwas) developed a tab-separated values (TSV) text format with a

66 minimal set of required (and optional) columns along with standardised headings [7]. The

67 SMR tool [8] introduced a binary format for rapid querying of quantitative trait loci. These

68 approaches are adequate for storing variant level summary statistics but do not enforce

69 allele consistency or support embedding of essential metadata. Learning from these

70 examples and our experiences performing high-throughput analyses across two research

71 centres, we developed a set of requirements for a suitable universal format (Table 1). These

72 features place emphasis on consistency and robustness, capacity for metadata to provide a

73 full audit trail, efficient querying and file storage, ensuring data integrity, interoperability

74 with existing open-source tools and across multiple datasets to support data sharing and

75 integration. We determined that adapting the variant call format (VCF) [9] was a convenient

76 and constructive solution to address these issues. We provide evidence demonstrating how

77 the VCF meets our requirements and showcase the capabilities of this medium (Table 1).

78

79   The VCF is organised into three components: a flexible file header containing metadata

80   (lines beginning with '#'), and a file body containing variant- (one locus per row with one or

81   more alternative alleles/variants) and sample-level information (one sample per column).

82   We adapt this format to include GWAS-specific metadata and utilise the sample column to

83   store variant-trait association data (Figure 1; Supplementary Table 1).

84

85   According to the VCF specification, the file header consists of metadata lines containing 1)

86   the specification version number, 2) information about the reference genome assembly and

87   contigs, and 3) information (ID, number, type, description, source and version) about the

88   fields used to describe variants and samples (or variant-trait associations in the case of

89   GWAS-VCF) in the file body. We take advantage of the VCF file header to store additional

90   information about the GWAS including 1) source and date of summary statistics, 2) study

91   IDs (e.g., PMID/DOI of publication describing the study, or accession number and repository

92   of individual-level data), 3) description of the trait(s) studied (e.g., type, association test

93   used, sample size, ancestry and measurement unit) as well as the source and version of trait

94   IDs (e.g., Experimental Factor Ontology [17], Human Phenotyping Ontology [18] or Medical

95   Subject Headings [19] IDs for clinical and other traits, or Ensembl Gene IDs for eQTL datasets).

96

97   Unlike VCF where a row can contain information about multiple alternative alleles observed

98   at the same site/locus (and thus may store more than one variant), the GWAS-VCF

99   specification requires that each variant is stored in a separate row of the file body. Each row

100   contains eight mandatory fields: chromosome name (CHROM), base-pair position (POS),

101   unique variant identifier (ID), reference/non-effect allele (REF), alternative/effect allele

102   (ALT), quality (QUAL), filter (FILTER) and variant information (INFO). The ID, QUAL and

103   FILTER fields can contain a null value represented by a dot. Importantly, the ID value (unless

104   null) should not be present in more than one row. The FILTER field may be used to flag poor

105   quality variants for exclusion in downstream analyses. The INFO column is a flexible data

106   store for additional variant-level key-value pairs (fields) and may be used to store for

107   example: population frequency (AF), genomic annotations and variant functional effects.

108   We also use the INFO field to store the dbSNP [20] locus identifier (rsid) for the site at which

109   the variant resides. This is because (despite their common usage as variant identifiers) rsids

110   uniquely identify loci (not variants!) and thus cannot be used in the ID field, as we will

111   discuss further at the end of this manuscript. Following the INFO column is a format field

112   (FORMAT) and one or more sample columns which we use to store variant-trait association

113   data, with values for the fields listed in the FORMAT column for example: effect size (ES),

114   standard error (SE) and -log10 P-value (LP).

115

116   This format has a number of advantages over existing solutions. First, the VCF provides

117   consistent and robust approaches to storing genetic variants, annotations and metadata.

118   Furthermore, variable type and number requirements reduce parsing errors and missing

119   data and prevent unexpected program operation. Second, the VCF is well established and

120   supported by existing tools providing a range of functions for querying, annotating,

121   transforming and analysing genetic data. Third, the GWAS-VCF file header stores

122   comprehensive metadata about the GWAS. Fourth, a GWAS-VCF file can store individual or

123   multiple traits (in one or more sample columns) in a single file which is beneficial for the

124   distribution of GWAS datasets where genotypes of each sample/individual have been tested

125   for association with multiple traits (e.g., eQTL datasets).

126

127 Simulations of query performance demonstrate compressed GWAS-VCF is substantially

128 quicker than unindexed and uncompressed TSV format for querying by genomic position.

129 On average GWAS-VCF was 16x faster to extract a single variant using chromosome position

130 (mean query duration in GWAS-VCF 0.08 seconds [95% CI 0.08, 0.08]) vs mean query

131 duration in TSV 1.29 seconds [95% CI 1.29, 1.30]) and 9x quicker using the rsid (0.09 seconds

132 [95% CI 0.09, 0.09] vs 0.81 seconds [95% 0.80, 0.82]). Using a 1Mb window of variants

133 GWAS-VCF was 46x quicker (0.11 seconds [95% CI 0.11, 0.11] vs 5.02 seconds [95% CI 4.99,

134 5.04]). Although querying on association P value was faster using TSV (mean query duration

135 in TSV 7.18 seconds [95% CI 7.09, 7.26] vs mean query duration in GWAS-VCF 18.04 seconds

136 [95% CI 17.92, 18.16]) GWAS-VCF could be improved by using variant flags (i.e. in the INFO

137 field) to highlight records below prespecified thresholds if the exact value is unimportant.

138 For example, all variants below genome-wide significance (P < 5e-8) or a more relaxed

139 threshold (e.g. P < 5e-5).

140

141 To automate the conversion of existing summary statistics files to the GWAS-VCF format, we

142 developed open-source Python3 software (Gwas2VCF; Table 2). The application reads in

143 metadata and variant-trait association data using a user-defined schema. During processing,

144 variants are harmonised using a supplied reference genome file to ensure the non-effect

145 allele matches the reference sequence enabling consistent directionality of allelic effects

146 across studies. Insertion-deletion variants are left-aligned and trimmed for consistent

147 representation using the vgraph library [21]. Finally, the GWAS-VCF is indexed using tabix [22]

148 and rsidx [23] which enable rapid queries by genomic position and rsid , respectively. We have

149 developed a freely available web application providing a user-friendly interface for this

150 implementation and encourage other centres to deploy their own instance (Table 2).

151

152   Once stored in a GWAS-VCF file, summary statistics can be read and queried using R or

153   Python programming languages with our open-source libraries (Table 2) or from the

154   command line using for example: bcftools [24], GATK [25] or bedtools [26]. Alternatively, GWAS-

155   VCF may be converted to NHGRI-EBI format [27] or any other tabular format to support

156   incompatible tools. Further, the gwasglue R package provides convenient programming

157   functions to automate preparation of genetic association data for a range of downstream

158   analyses (Table 2). Currently, methods exist for streamlining variant fine-mapping [28–32],

159   colocalization [33], MR [34] and data visualisation [35]. New methods are being actively added and

160   users may request new features via the repository issues page.

161

162   To encourage adoption, we made openly available over 10,000 complete GWAS summary

163   statistics in GWAS-VCF format as part of the IEU GWAS database. These studies include a

164   broad range of traits, diseases and molecular phenotypes building on the initial collection

165   for the MR Base platform [34].

166

167   A limitation of current summary statistics formats, including GWAS-VCF, is the lack of a

168   widely adopted and stable representation of sequence variants that can be used as

169   universal unique identifier for said variants. Published summary statistics often use rsids [20]

170   to identify variants but this practice is inappropriate because rsids are locus identifiers and

171   do not distinguish between multiple alternative alleles observed at the same site. Moreover,

172   rsids are not stable as they can be merged and retired over time. The reason this is a

173   problem is that in GWAS summary statistics every record represents the effect of a specific

174   allele on one or more traits, and if a record identifier is used that is not unique for each

175   allelic substitution it cannot technically be considered an identifier. An alternative approach

176   is to concatenate chromosome, base-position, reference and alternative allele field values

177   into a single string, but this is non-standardised, and genome build specific. Worst still is the

178   common approach of mixing these types of identifiers within a single file. In version 1.1 of

179   the GWAS-VCF specification we suggest querying variants by chromosome and base-

180   position and filtering the output to retain the target substitution (implemented in our

181   parsers), but we acknowledge that this approach can be cumbersome and difficult to

182   interoperate with other software. The ideal solution would be to populate the ID column of

183   a GWAS-VCF file using universally accepted and unique variant identifiers. We have

184   reviewed several existing variant identifier formats as candidates for the variant identifier

185   field, to be implemented in the next version of the specification (Supplementary Table 2).

186   However, we refrain from making a unilateral choice at this juncture because successful

187   implementation will require consultation from a range of stakeholders. The genetics

188   community uses different approaches already to deal with the problem of sequence variant

189   representation and there is a need to coalesce upon a single format.

190

191   Here we present an adaptation of the VCF specification for GWAS summary statistics

192   storage that is amenable to high-throughput analyses and robust data sharing and

193   integration. We implement open-source tools to convert existing summary statistics formats

194   to GWAS-VCF, and libraries for reading or querying this format and integrating with existing

195   analysis tools. Finally, we provide complete GWAS summary statistics for over 10,000 traits

196   in GWAS-VCF. These resources enable convenient and efficient secondary analyses of GWAS

197   summary statistics and support future tool development.

198

199   **Code availability**

200

201   Open-source query performance evaluation source code available from GitHub

202   (https://github.com/MRCIEU/gwas-vcf-performance) or pre-built image available from

203   DockerHub (mrcieu/gwas-vcf-performance)

204

205   **Data availability**

206

207   Version 1.1 of the GWAS -VCF format specification is available from:

208   https://github.com/MRCIEU/gwas-vcf-spec/releases/tag/1.1

209

210   Full summary statistics for over 10,000 GWAS in VCF format are available from the IEU

211   GWAS Database (https://gwas.mrcieu.ac.uk)

212

213   **Method**

214

215   **Specification**

216

217   The specification was developed through experience of collecting and harmonising GWAS

218   summary data across two research centres at scale [34] and performing a range of

219   representative high throughput analyses on these data (for example LD score regression [36],

220   MR [37], genetic colocalisation analysis [38] and polygenic risk scores [39]).

221

222   **Query performance simulation**

223

224   Densely imputed summary statistics (13,791,467 variants) for a large GWAS of body mass

225   index data were obtained from Neale et al [40]. The data were mapped to VCF using

226   Gwas2VCF v1.1.1 and processed using bcftools v1.10 [24] to remove multiallelic variants or

227   records with missing dbSNP [20] identifiers. A tabular (unindexed) file was prepared from the

228   VCF to replicate a typical storage medium currently used for distributing summary statistics.

229   Query runtime performance was compared between tabix v1.10.2 [22] and standard UNIX

230   commands under the following conditions: single variant selection using dbSNP identifier [20]

231   or chromosome position, multi-variant selection by association P value (thresholds: P < 5e-

232   8, 0.2, 0.4, 0.6, 0.8) or 1 Mb genomic interval. Tests were undertaken with 100 repetitions

233   using VCF or unindexed text formats with and without GZIP compression on an Ubuntu

234   v18.04 server with Intel Xeon(R) 2.0 Ghz processor. All comparisons were performed using

235   singled thread operations and therefore differences in runtime performance were due to

236   tool and/or file index usage.

237

238   **References**

239

240   1.   Hou, L. & Zhao, H. A review of post-GWAS prioritization approaches. *Front. Genet.* **4**,

241   280 (2013).

242   2.   Finucane, H. K. *et al.* Partitioning heritability by functional annotation using genome-

243   wide association summary statistics. *Nat. Genet.* **47**, 1228–1235 (2015).

244   3.   Visscher, P. M. *et al.* 10 Years of GWAS Discovery: Biology, Function, and Translation.

245   *American Journal of Human Genetics* **101**, 5–22 (2017).

246   4.   Smith, G. D. & Ebrahim, S. 'Mendelian randomization': Can genetic epidemiology

247   contribute to understanding environmental determinants of disease? *International*

248   *Journal of Epidemiology* (2003). doi:10.1093/ije/dyg070

249   5.   Bulik-Sullivan, B. *et al.* LD score regression distinguishes confounding from

250   polygenicity in genome-wide association studies. *Nat. Genet.* (2015).

251   doi:10.1038/ng.3211

252   6.   Yang, J., Zeng, J., Goddard, M. E., Wray, N. R. & Visscher, P. M. Concepts, estimation

253   and interpretation of SNP-based heritability. *Nature Genetics* **49**, 1304–1310 (2017).

254   7.   Buniello, A. *et al.* The NHGRI-EBI GWAS Catalog of published genome-wide

255   association studies, targeted arrays and summary statistics 2019. *Nucleic Acids Res.*

256   **47**, D1005–D1012 (2019).

257   8.   Zhu, Z. *et al.* Integration of summary data from GWAS and eQTL studies predicts

258   complex trait gene targets. *Nat. Genet.* **48**, 481–487 (2016).

259   9.   Danecek, P. *et al.* The variant call format and VCFtools. *Bioinformatics* **27**, 2156–2158

260   (2011).

261   10.   Purcell, S. *et al.* PLINK: A tool set for whole-genome association and population-based

262   linkage analyses. *Am. J. Hum. Genet.* (2007). doi:10.1086/519795

263   11.   Yang, J., Lee, S. H., Goddard, M. E. & Visscher, P. M. GCTA: A tool for genome-wide

264   complex trait analysis. *Am. J. Hum. Genet.* **88**, 76–82 (2011).

265   12.   Loh, P. R. *et al.* Efficient Bayesian mixed-model analysis increases association power

266   in large cohorts. *Nat. Genet.* **47**, 284–290 (2015).

267   13.   Zhou, X. & Stephens, M. Genome-wide efficient mixed-model analysis for association

268   studies. *Nat. Genet.* **44**, 821–824 (2012).

269   14.   Shabalin, A. A. Gene expression Matrix eQTL: ultra fast eQTL analysis via large matrix

270   operations. **28**, 1353–1358 (2012).

271   15.   Willer, C. J., Li, Y. & Abecasis, G. R. METAL: fast and efficient meta-analysis of

272   genomewide association scans. *Bioinforma. Appl. NOTE* **26**, 2190–2191 (2010).

273   16.   Hartwig, F. P., Davies, N. M., Hemani, G. & Smith, G. D. Two-sample Mendelian

274   randomization: avoiding the downsides of a powerful, widely applicable but

275   potentially fallible technique. *Int. J. Epidemiol.* 1717–1726 (2016).

276   doi:10.1093/ije/dyx028

277   17.   Malone, J. *et al.* Databases and ontologies Modeling sample variables with an

278   Experimental Factor Ontology. **26**, 1112–1118 (2010).

279   18.   Köhler, S. *et al.* Expansion of the Human Phenotype Ontology (HPO) knowledge base

280   and resources. *Nucleic Acids Res.* **47**, D1018–D1027 (2019).

281   19.   Medical Subject Headings - Home Page. Available at:

282   https://www.nlm.nih.gov/mesh/meshhome.html. (Accessed: 16th April 2020)

283   20.   Sherry, S. T. *et al. dbSNP: the NCBI database of genetic variation. Nucleic Acids*

284   *Research* **29**, (2001).

285   21.   bioinformed/vgraph: vgraph is a command line application and Python library to

286   compare genetic variants using variant graphs. ``vgraph`` utilizes a graph

287   representation of genomic variants in to precisely compare complex variants that are

288   refractory to comparison by conventional comparison methods. Available at:

289   https://github.com/bioinformed/vgraph. (Accessed: 5th May 2020)

290   22.   Li, H. Tabix: fast retrieval of sequence features from generic TAB-delimited files.

291   *Bioinforma. Appl. NOTE* **27**, 718–719 (2011).

292   23.   bioforensics/rsidx: Library for indexing VCF files for random access searches by rsID.

293   Available at: https://github.com/bioforensics/rsidx. (Accessed: 5th March 2020)

294   24.   Li, H. A statistical framework for SNP calling, mutation discovery, association mapping

295 and population genetical parameter estimation from sequencing data. *Bioinformatics*

296 **27**, 2987–93 (2011).

297 25. McKenna, A. *et al.* The genome analysis toolkit: A MapReduce framework for

298 analyzing next-generation DNA sequencing data. *Genome Res.* (2010).

299 doi:10.1101/gr.107524.110

300 26. Quinlan, A. R. & Hall, I. M. BEDTools: a flexible suite of utilities for comparing genomic

301 features. *Bioinforma. Appl. NOTE* **26**, 841–842 (2010).

302 27. MacArthur, J. *et al.* The new NHGRI-EBI Catalog of published genome-wide

303 association studies (GWAS Catalog). *Nucleic Acids Res.* **45**, D896–D901 (2017).

304 28. Benner, C. *et al.* Genetics and population analysis FINEMAP: efficient variable

305 selection using summary data from genome-wide association studies.

306 doi:10.1093/bioinformatics/btw018

307 29. Kichaev, G. *et al.* Integrating Functional Data to Prioritize Causal Variants in Statistical

308 Fine-Mapping Studies. *PLoS Genet.* **10**, e1004722 (2014).

309 30. Kichaev, G. & Pasaniuc, B. Leveraging Functional-Annotation Data in Trans-ethnic

310 Fine-Mapping Studies. *Am. J. Hum. Genet.* **97**, 260–271 (2015).

311 31. Kichaev, G. *et al.* Improved methods for multi-trait fine mapping of pleiotropic risk

312 loci. *Bioinformatics* **33**, 248–255 (2017).

313 32. Hormozdiari, F., Kostem, E., Kang, E. Y., Pasaniuc, B. & Eskin, E. Identifying causal

314 variants at loci with multiple signals of association. *Genetics* **198**, 497–508 (2014).

315 33. Wallace, C. Statistical Testing of Shared Genetic Control for Potentially Related Traits.

316 *Genet. Epidemiol.* **37**, 802–813 (2013).

317 34. Hemani, G. *et al.* The MR-base platform supports systematic causal inference across

318 the human phenome. *Elife* **7**, (2018).

319 35. jrs95/gassocplot: Regional association plotter for genetic and epigenetic data.

320   Available at: https://github.com/jrs95/gassocplot. (Accessed: 21st April 2020)

321 36. Zheng, J. *et al.* Databases and ontologies LD Hub: a centralized database and web

322   interface to perform LD score regression that maximizes the potential of summary

323   level GWAS data for SNP heritability and genetic correlation analysis. *Bioinformatics*

324   **33**, 272–279 (2017).

325 37. Hemani, G. *et al.* Automating Mendelian randomization through machine learning to

326   construct a putative causal map of the human phenome. *bioRxiv* 173682. (2017).

327   doi:10.1101/173682

328 38. Richardson, T. G., Hemani, G., Gaunt, T. R., Relton, C. L. & Davey Smith, G. A

329   transcriptome-wide Mendelian randomization study to uncover tissue-dependent

330   regulatory mechanisms across the human phenome. *Nat. Commun.* **11**, 1–11 (2020).

331 39. Richardson, T. G., Harrison, S., Hemani, G. & Smith, G. D. An atlas of polygenic risk

332   score associations to highlight putative causal relationships across the human

333   phenome. *Elife* **8**, (2019).

334 40. UK Biobank — Neale lab. Available at: http://www.nealelab.is/uk-biobank/.

335   (Accessed: 25th February 2020)

336 41. Li, H. *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinforma. Appl.*

337   *NOTE* **25**, 2078–2079 (2009).

338 42. Obenchain, V. *et al.* Sequence analysis VariantAnnotation: a Bioconductor package

339   for exploration and annotation of genetic variants. **30**, 2076–2078 (2014).

340 43. Gentleman, R. C. *et al. Open Access Bioconductor: open software development for*

341   *computational biology and bioinformatics. Genome Biology* **5**, (2004).

342 44. Huber, W. *et al.* Orchestrating high-throughput genomic analysis with Bioconductor.

Nat. Methods **12**, 115–121 (2015).

45.   Bioconductor - Home. Available at: https://www.bioconductor.org/. (Accessed: 27th March 2020)

46.   pysam-developers/pysam: Pysam is a Python module for reading and manipulating SAM/BAM/VCF/BCF files. It's a lightweight wrapper of the htslib C-API, the same one that powers samtools, bcftools, and tabix. Available at: https://github.com/pysam-developers/pysam. (Accessed: 10th March 2020)

47.   IEU GWAS database. Available at: https://gwas.mrcieu.ac.uk/. (Accessed: 10th March 2020)

48.   broadinstitute/picard: A set of command line tools (in Java) for manipulating high-throughput sequencing (HTS) data and formats such as SAM/BAM/CRAM and VCF. Available at: https://github.com/broadinstitute/picard. (Accessed: 25th February 2020)

49.   GenomicsDB/GenomicsDB: Highly performant data storage in C++ for importing, querying and transforming variant data with Java/Spark. Used in gatk4. Available at: https://github.com/GenomicsDB/GenomicsDB. (Accessed: 25th February 2020)

50.   Voss, K., Gentry, J. & Auwera, G. Van Der. GATK4 + WDL + Cromwell. *F1000Research* **6**, 4 (2017).

51.   Morales, J. *et al.* A standardized framework for representation of ancestry data in genomics studies, with application to the NHGRI-EBI GWAS Catalog. *Genome Biol.* **19**, 21 (2018).

52.   den Dunnen, J. T. *et al.* HGVS Recommendations for the Description of Sequence Variants: 2016 Update. *Hum. Mutat.* **37**, 564–569 (2016).

53.   Holmes, J. B., Moyer, E., Phan, L., Maglott, D. & Kattman, B. SPDI: data model for

367   variants and applications at NCBI. doi:10.1093/bioinformatics/btz856

368   54.   Wagner, A. *et al.* ga4gh/vr-spec: 1.0 GA4GH Approved. (2019).

369   doi:10.5281/ZENODO.3572974

370

Acknowledgments

371   **Acknowledgments**

372

385

386   **Author contributions**

387

388   All authors contributed the manuscript and storage format specification. G.H. and E.M.

389   designed the research. M.L. and G.H. wrote software packages and performed query

390   performance simulations. B.E. and G.H. prepared the GWAS data.

391

Competing interest

393

394   None.

395

Correspondence

397

398   Matthew Lyon (matt.lyon@bristol.ac.uk)

399   Population Health Sciences

400   Bristol Medical School

401   University of Bristol

402   Oakfield House

403   Oakfield Grove

404   Bristol

405   BS8 2BN

Table 1. Requirements for a summary statistics storage format and solutions offered by the VCF

Requirement	Solution using the variant call format
Human readable and easy to parse	Easily read with any text viewer. Mature open-source parsing libraries are available (HTSLIB [41] and HTSJDK [41]) and implemented in most modern programming languages, for example: VariantAnnotation [42] R-package is available from Bioconductor [43–45] and python package pysam [46]. Bcftools [24], GATK [25], bedtools [26] and others provides user-friendly functionality from the command line.
Unambiguous interpretation of the data	Data field descriptions, value types and number of values are required and defined in the file header. File validity is enforced during each read/write.
Unambiguous representation of bi-allelic, multi-allelic and insertion-deletion variants	Every variant substitution is represented by reference and alternative allele haplotypes defining the exact base change on the forward strand. The reference allele is required to match genome sequences defined in the file header. The alternative allele is always the effect allele allowing consistency between studies for ease of comparison.
Genomic information can be validated	The file header contains information about reference genome assembly and contigs. Reference alleles must match the sequence in the referenced genome build (in FASTA format). GATK [25] ValidateVariants can be used to verify file format validity and compare reference allele information against the corresponding genome reference sequence.
Flexibility on which GWAS fields are recorded and enforcement of essential fields	All fields are defined in the file header and can be set optional or required as desired. The specification contains essential fields and their reserved names.
Capacity to store metadata about the study and trait(s)	The file header contains information about the source and date of summary statistics, study IDs (e.g., PMID/DOI of publication describing the study, or accession number and repository of individual-level data), description of the trait(s) studied (e.g., type, association test used, and measurement unit) as well as the source and version of trait IDs (e.g., IEU GWAS database [47], Experimental Factor Ontology [17], Human Phenotyping Ontology [18] or Medical Subject Headings [19] IDs for clinical and other traits, or Ensembl Gene IDs for eQTL datasets).
Allows multiple traits to be stored together	The SAMPLE column was chosen to store variant-trait association data to allow for storage of multiple traits in a single VCF file, or as individual files if desired.
Rapid querying by variant identifier, genomic position interval or GWAS	The file is sorted karyotypically and indexed by chromosome position using tabix [22] to enable fast queries by genomic position. Secondary indexing on dbSNP [20] identifier is also provided using rsidx [23]. Refer to performance comparisons of indexed VCF files and standard UNIX tools.

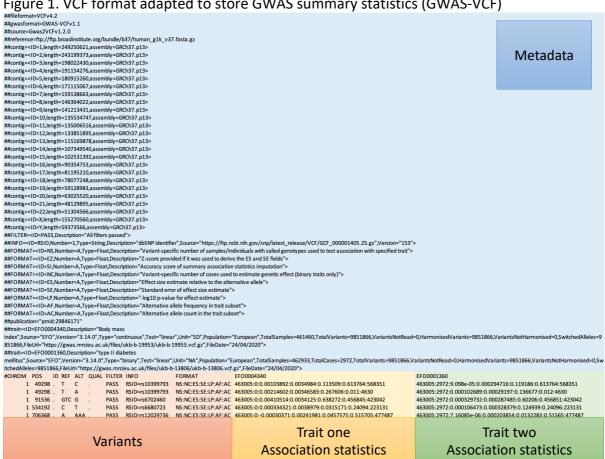|	
summary statistics value (range or exact value)	
File compression	VCF files may be compressed with block GZIP [24] or converted to a binary call file which is a binary VCF companion format [24].
Readable by existing open-source tools	A large number of tools support VCF files including: GATK [25], Picard [48], bcftools [24], bedtools [26], vcftools [9] and plink [10]. Bcftools [24] can also provide a tabular extract for use with non-compatible tools.
Amenable to cloud-based streaming and database storage	Genomic intervals may be extracted over a network using a range-request which extracts file segments without transferring the whole file. This enables rapid streaming of queries over the internet. For high-throughput and distributed storage and querying, VCF files can be easily imported into GenomicsDB [49].

GWAS, genome-wide association study. dbSNP, database of single-nucleotide polymorphisms. HTSLIB, high-throughput sequencing data library. HTSJDK, high-throughput sequencing data java development kit. GATK, genome-analysis toolkit. dbSNP, single nucleotide polymorphism database. eQTL, expression quantitative trait loci.

Table 2. Open-source tools for working with GWAS-VCF

Program	Purpose	Implementation	Source code link
gwas2vcf	Mapping tabular GWAS summary statistics and NHGRI-EBI format to VCF	Python3 (Docker)	https://github.com/mrcieu/gwas2vcf
gwas2vcfweb http://vcf.mrcieu.ac.uk	Front-end and queue schedular for gwas2vcf	Python3, Cromwell [50] (Docker)	https://github.com/mrcieu/gwas2vcfweb
R/gwasvcf	Library for querying and reading GWAS-VCF files	R	https://github.com/mrcieu/gwasvcf
pygwasvcf	Library for querying and reading GWAS-VCF files	Python3	https://github.com/mrcieu/pygwasvcf
R/gwasglue	Library for processing GWAS summary statistics ready for secondary analysis	R	https://github.com/mrcieu/gwasglue
LD Score Regression [5] **(patch)**	Estimating genetic correlation and heritability	Python	http://github.com/explodecomputer/ldsc

GWAS, genome-wide association study. LD, linkage disequilibrium. VCF, variant call format. NHGRI-EBI, National Human Genome Research Institute and European Bioinformatics Institute.

Figure 1. VCF format adapted to store GWAS summary statistics (GWAS-VCF)

The GWAS-VCF file contains study and trait(s) metadata, variant-level data, and variant-trait association summary statistics. Each field is defined in the file header including variable type and number of values. The format can store the results of a GWAS with one or more traits in a single file.

Figure 2. Performance comparison for querying summary statistics in plain text and GWAS-VCF

Query runtime of VCF and unindexed text using a range of operations

Mean query time (log milliseconds [lower is quicker]; repetitions n=100) to extract either: a single variant using the chromosome position or dbSNP [20] identifier or multiple variants using a 1 Mb interval or association P value. AWK, grep, bcftools [24] and rsidx [23] were evaluated using uncompressed and GZIP/BGZIP [24] compressed unindexed text and VCF. Error bars represent the 95% confidence interval.

Supplementary Table 1. Data fields in the GWAS-VCF

Field	Description
VCF Header	
Publication	Reference to publication describing the study in compact uniform resource identifier (CURIE) format (prefix:reference) e.g. doi:10.1000/xyz123 or pmid:12345678
Trait ID*	Trait identifier e.g. an ontology or metadata repository identifier e.g. EFO0004340 (EFO) or ieu-a-835 (IGD)
Description	Trait description e.g. Body mass index
Source	Source of trait identifier e.g. EFO [17] or IGD [47]
Version	Version of trait ID source used to describe trait
Type	Outcome variable type (continuous or binary)
Test	Statistical test for association data e.g. linear regression
Unit	Phenotype units e.g. kg/m2 or SD
Population	Participant ancestry (or mixed ancestry) using the standardised framework [51]
FileUrl	URL of GWAS summary statistics file
FileDate	Date GWAS summary statistics were produced
TotalSamples	Total number of samples/individuals in the study
TotalCases	Total number of cases in the study (if case-control)
TotalVariants	Total number of variants tested in the study
VariantsNotRead	Number of variants that could not be read
VariantsHarmonised	Number of harmonised variants
VariantsNotHarmonised	Number of variants that could not be harmonised
SwitchedAlleles	Number of variants strand switched
VCF FORMAT (per trait variant-level information)	
NS	Variant-specific number of samples/individuals with called genotypes used to test association with specified trait
EZ	Z-score provided if it was used to derive the ES and SE fields
SI	Accuracy score of association statistics imputation
NC	Variant-specific number of cases used to estimate genetic effect (binary traits only)
ES*	Effect size estimate relative to the alternative allele
SE*	Standard error of effect size estimate
LP*	-log10 p-value for effect estimate
AF	Alternative allele frequency in trait subset
AC	Alternative allele count in the trait subset

ID, identifier. EFO, Experimental Factor Ontology. IGD, IEU GWAS database. * Required fields.

Supplementary Table 2. Possible variant identifier schemes for the ID column of GWAS-VCF

VCF row identifier (ID column)	Advantages	Disadvantages
dbSNP [20] rsID with multiallelic variants on a single row **Example:** **rs376272854**	• No duplication of information already in the row • Rsidx [23] provides fast dbSNP [20] ID queries • Widely used • Short length • Compatibility with existing tools (rsid is encouraged by VCF [9] v4.2 specification)	• Refers to a position rather than a substitution • Complexity and ambiguity of manipulating multiallelic rows • Does not distinguish between multiple alternative alleles and therefore a positional identifier • Multiple rsids can point to the same position (e.g. new dbSNP [20] entries awaiting merge with existing records)
No value in ID column with multiallelic variants on separate rows	• No duplication of information already in the row • Avoids the complexities of a variant identifier	• Variant queries include multiple fields (chromosome, position, reference and alternative allele) • No guarantees of row uniqueness • Difficult to operate with other software that requires a unique substitution identifier
HGVS [52] DNA nomenclature with multiallelic variants on separate rows **Example:** **chr2:g.84918761_84918811del**	• Unique identifier for every substitution • Supports one substitution per row in the VCF which is easier to parse • Short insertion-deletion encoding • Known format	• Duplicates information already stored in the row • Not stable between genome builds • Comparing between builds is difficult • Not widely used for GWAS
Concatenation of chromosome, position and alleles with multiallelic variants on separate rows **Example:** **chr2:84918760:**	• Unique identifier for every substitution • Supports one substitution per row in the VCF which is easier to parse • Known format	• Duplicates information already stored in the row • Comparing between builds is difficult • Not stable between genome builds • Long insertion-deletion coding

CCCAACCCTGCTGTCAT AATGCATAAGCAGCCAC AGACAGTAAGTGAATGAA:C		
SPDI [53] (Sequence-id, Position, Deleted Sequence, Insertion Sequence separated by a colon) with multiallelic variants on separate rows **Example:** **NC_000002.12: 84918760: CCCAACCCTGCTGTCAT AATGCATAAGCAGCCAC AGACAGTAAGTGAATGAA:C**	• Unique identifier for every substitution • Supports one substitution per row in the VCF which is easier to parse Known format	• Duplicates information already stored in the row • Comparing between builds is difficult • Not stable between genome builds • Long insertion-deletion coding
Concatenation of chromosome, position and alleles using MD5 hash to shorten long alleles with multiallelic variants on separate rows **Example:** **chr2:84918760- 7c43e7284b58ba06e 7438bff62376edf:C**	• Unique (almost) identifier for every substitution • Supports one substitution per row in the VCF which is easier to parse • Short insertion-deletion coding	• Duplicates information already stored in the row • Not stable between genome builds • Comparing between builds is difficult • Cannot reverse hash without database • Not widely used
GA4GH Variation Representation [54] (SHA-512 message digest of the chromosome position and alternative allele with	• Unique (almost) identifier for every substitution • Supports one substitution per row in the VCF which is easier to parse Short insertion-deletion coding	• Duplicates information already stored in the row • Not stable between genome builds • Comparing between builds is difficult • Cannot reverse hash without database

multiallelic variants on separate rows **Example:** **ga4gh:VA.yOoxi7-uUnJyn4QkQ23h6RJuT4Zqarow**		• Not widely used

GWAS, genome-wide association study. VCF, variant call format. Rsidx, file index using the dbSNP identifier. MD5, message-digest algorithm. HGVS, Human Genome Variation Society. GA4GH, Global Alliance for Genomics and Health. SHA, Secure Hash Algorithm.