Method

Ultrafast and scalable variant annotation and prioritization with big functional genomics data

Dandan Huang,^{1,2,3} Xianfu Yi,⁴ Yao Zhou,² Hongcheng Yao,⁵ Hang Xu,^{1,5} Jianhua Wang,² Shijie Zhang,² Wenyan Nong,⁶ Panwen Wang,⁷ Lei Shi,³ Chenghao Xuan,³ Miaoxin Li,⁸ Junwen Wang,⁷ Weidong Li,⁹ Hoi Shan Kwan,⁶ Pak Chung Sham,¹⁰ Kai Wang,¹¹ and Mulin Jun Li^{1,2,12}

¹ The Province and Ministry Co-sponsored Collaborative Innovation Center for Medical Epigenetics, National Clinical Research Center for Cancer, Tianjin Medical University Cancer Institute and Hospital, Tianjin Medical University, Tianjin 300070, China; ² Department of Pharmacology, Tianjin Key Laboratory of Inflammation Biology, School of Basic Medical Sciences, Tianjin Medical University, Tianjin 300070, China; ³ Department of Biochemistry and Molecular Biology, School of Basic Medical Sciences, Tianjin Medical University, Tianjin 300070, China; ⁴ School of Biomedical Engineering, Tianjin Medical University, Tianjin 300070, China; ⁵ School of Biomedical Sciences, LKS Faculty of Medicine, The University of Hong Kong, Hong Kong SAR 999077, China; ⁶ School of Life Sciences, The Chinese University of Hong Kong, Hong Kong SAR 999077, China; ⁷ Department of Health Sciences Research and Center for Individualized Medicine, Mayo Clinic, Scottsdale, Arizona 85259, USA; ⁸ Center for Genome Research, Center for Precision Medicine, Zhongshan School of Medicine, First Affiliated Hospital, Sun Yat-Sen University, Guangzhou 510080, China; ⁹ Department of Genetics, School of Basic Medical Sciences, Tianjin Medical University, Tianjin 300070, China; ¹⁰ Centre of Genomics Sciences, Departments of Psychiatry, LKS Faculty of Medicine, The University of Hong Kong, Hong Kong SAR 999077, China; ¹¹ Raymond G. Perelman Center for Cellular and Molecular Therapeutics, Children's Hospital of Philadelphia, Philadelphia, Pennsylvania 19104, USA; ¹² Department of Epidemiology and Biostatistics, Tianjin Key Laboratory of Molecular Cancer Epidemiology, Tianjin Medical University Cancer Institute and Hospital, Tianjin Medical University, Tianjin 300070, China

The advances of large-scale genomics studies have enabled compilation of cell type–specific, genome-wide DNA functional elements at high resolution. With the growing volume of functional annotation data and sequencing variants, existing variant annotation algorithms lack the efficiency and scalability to process big genomic data, particularly when annotating whole-genome sequencing variants against a huge database with billions of genomic features. Here, we develop VarNote to rapidly annotate genome-scale variants in large and complex functional annotation resources. Equipped with a novel index system and a parallel random-sweep searching algorithm, VarNote shows substantial performance improvements (two to three orders of magnitude) over existing algorithms at different scales. It supports both region-based and allele-specific annotations and introduces advanced functions for the flexible extraction of annotations. By integrating massive base-wise and context-dependent annotations in the VarNote framework, we introduce three efficient and accurate pipelines to prioritize the causal regulatory variants for common diseases, Mendelian disorders, and cancers.

[Supplemental material is available for this article.]

Variant annotation is a common procedure in human genome studies for interpreting the biological function and disease relevance of given genetic variants or somatic mutations (MacArthur et al. 2014). It greatly aids prioritization analysis regarding certain genetic hypotheses and facilitates functional follow-up on selected variants. As the growing volume of large-scale genome sequencing of human populations, such as the UK Biobank study (Bycroft et al. 2018), the National Heart, Lung, and Blood Institute's Trans-Omics for Precision Medicine (TOPMed) program (Brody et al. 2017), and functional genomics data, such as The Encyclopedia of DNA Elements (ENCODE) Project (The ENCODE Project Consortium 2012; Davis et al. 2018) and the International Human Epigenome Consortium (IHEC) project (Bujold et al. 2016), efficient interpretation of genome variants is profoundly af-

Corresponding author: mulinli@connect.hku.hk

Article published online before print. Article, supplemental material, and publication date are at http://www.genome.org/cgi/doi/10.1101/gr.267997.120.

fected by the unprecedented scale of genomic features and the resources used for their annotation. For example, a widely used metric for mutation deleteriousness, Combined Annotation Dependent Depletion (CADD) (Kircher et al. 2014), integrates more than 100 annotations for all 8.6 billion possible substitutions and 48 million short indels in the human reference genome, and it is archived in a compressed file of >300 GB. Based on the whole CADD annotation database, it may take 5–100 h to finish a personal genome annotation (around 5 million single-nucleotide variants [SNVs] and indels) using present state-of-the-art variant annotation tools. Another resource, CistromeDB (Zheng et al. 2019), aggregates 360 million genomic intervals for more than 6000 human tissue-/cell type-specific epigenomic profiles; such

© 2020 Huang et al. This article is distributed exclusively by Cold Spring Harbor Laboratory Press for the first six months after the full-issue publication date (see http://genome.cshlp.org/site/misc/terms.xhtml). After six months, it is available under a Creative Commons License (Attribution-NonCommercial 4.0 International), as described at http://creativecommons.org/licenses/ by-nc/4.0/.

Huang et al.

cumulative region-based annotations will ultimately pose a challenge to the efficient interpretation of noncoding regulatory variants. Therefore, the development of fast, scalable, and versatile annotation retrieval strategies is crucial for the broad use of big genomic features in genetic study and precision medicine.

To retrieve relevant information from annotation databases, computational methods need first to identify annotation records that overlap query variants and then extract specified annotation fields. Such processing is usually done by sequential chromosome sweeping between query variants and the annotation databaseusing such as BEDTools (Quinlan and Hall 2010), BEDOPS (Neph et al. 2012), ANNOVAR (Wang et al. 2010), and BCFtools (Li 2011a)-or independent random access from a whole annotation database using an index, such as the UCSC binning algorithm (Kent et al. 2002), Tabix (Li 2011b) and VEP (McLaren et al. 2016). The vcfanno annotator implements a parallel chrom-sweep algorithm based on a Tabix index and enables streaming query within defined chunks (Pedersen et al. 2016). Recently, GIGGLE introduced a temporal indexing scheme for the fast identification of shared genomic loci between query features and thousands of genome interval files (Layer et al. 2018). Despite their achievements, no algorithm can properly adapt to query variants and annotation databases with different data scales and levels of feature distribution. This unresolved bottleneck can be attributed to the high burden of disk reads from massive annotation records, which significantly hamper the running speed and scalability of existing tools. In addition, few tools have been developed to efficiently accommodate genome-scale queries and accurately prioritize disease-causal variants by leveraging large-scale functional genomics data.

Here, we present a novel index system and an ultrafast parallel intersection algorithm, called VarNote, to process variant annotations and genomic features at scale. VarNote fits different levels of data distribution and reduces the runtime of common variant annotation tasks by more than two to three orders of magnitude. It supports both region-based and allele-specific annotations for different file formats and introduces many advanced functions to improve its flexibility in use. To facilitate efficient and accurate prioritization of disease-causal regulatory variants for clinicians and biologists in different medical genetics fields, we also develop three online VarNote applications: (1) causal regulatory variants prioritization from GWAS results of common diseases; (2) pathogenic regulatory variants prioritization from genome sequencing of rare inherited diseases; and (3) driver regulatory variants prioritization from personal genome profile of cancers.

Results

VarNote index system and random-sweep algorithm

Functional genomics studies constantly produce unprecedentedly large amounts of data at genome-wide scale, which enables comprehensive annotation of genetic variants. A Tabix index together with an associated bgzip file is currently one of the most widely used storage formats for genomic annotation. Tabix combined the binning index and linear index to quickly retrieve features overlapping specified regions, but it was optimized for single independent queries. For tasks that involve the query of each of the whole-genome sequencing (WGS) variants over a huge annotation database (e.g., CADD contains more than 100 annotations for each of 8.6 billion possible substitutions), tools relying on a Tabix index need to repetitively decompress gzip blocks and parse chromosome positions from original annotation records, thus introducing many redundant operations that can be reduced by more efficient means.

To maximally reduce time-consuming disk reads, we developed a novel index system for annotation databases. Given a bgzip-compressed annotation database, we created a positioning file that only keeps query-dependent information and an index file that allows fast retrieval of genomic position (Fig. 1A; Supplemental Fig. S1). Briefly, we first tailored the annotation metadata of each record in the original compressed block (OB) and used a reduced virtual block (ROB) that only stores block summary information together with chromosome position information for each record (Supplemental Fig. S2A). To further compress the record position information, we introduced an 8-bit "RecordFlag" to dynamically determine the exact storage volume of the chromosome position and block offset for sequential records (Fig. 1B). These strategies reduce the size of the original annotation database by a factor of approximately 100. For instance, the algorithm can convert 344 GB of CADD annotations for all possible SNVs into a 6-GB VarNote positioning file. Building on this positioning file with lossless compression of query-dependent information, we created a linear index that merely contains summary information of each ROB (SROB) (Supplemental Figs. S2B, S3), which ensures memory-efficient sweeping of chromosome positions. Taken together, our VarNote index system will significantly minimize disk reads during annotation and provide infrastructure for the fast retrieval of large numbers of query intervals (for details, see Methods).

By leveraging the VarNote index system, we also combined random-access and chromosome-sweep strategies to implement a unified and efficient searching approach for sorted query intervals/variants, called the random-sweep algorithm (Fig. 1C). Specifically, the algorithm first loads the VarNote small index and sequentially sweeps SROBs to locate intersected ROBs. Selected ROBs are random accessed from the VarNote positioning file through block summary information, while unassociated ROBs are directly skipped. As position information for each annotation record is bit-encoded within consecutive ROBs, decoding ROB content and applying the chromosome-sweep algorithm can identify all annotation record hits (Supplemental Fig. S4; Methods).

The gained speed and scalability of VarNote were attained for the following reasons. First, the intersection between query intervals and annotation records mostly relies on the VarNote positioning file instead of the original annotation database. The only step associated with the original annotation database is the extraction of annotation fields for final record hits using random access, hence excessive disk reads can be largely saved. Second, randomsweep searching is a coherent process in which the combination of a global linked list and a file pointer ensure straightforward intersections without returns; meanwhile, the algorithm can jump over unrelated data blocks. This strategy enables VarNote to be scalable to large data sets and still be efficient for small inputs, especially for sparse and unbalanced queries. Third, there is no repetitive decomposition of the same gzip blocks during the annotation process, thus further accelerating annotation searches.

Comparison of VarNote with existing tools for interval-level annotations

Genomic features intersection is a common part of many bioinformatics analyses such as variant annotation, yet existing tools rely strongly on the data scales and feature distributions of the query



Figure 1. The key components of VarNote. (A) Architecture of VarNote index system. Bgzip-compressed annotation database (.bgz file) will be first converted to VarNote positioning file (.vanno file). The system tailors and encodes information of each original compressed block in the annotation database (OB) to generate a reduced virtual block that only keeps query-dependent information (ROB). The bgzip-compressed VarNote positioning file contains concatenated compressed block that stores ROB bytes. Then, summary information of the reduced virtual block (SROB) is linearly indexed to generate VarNote index file (.vi file). (B) Bit encoding of record position information. The system uses an 8-bit "RecordFlag" to encode position information of annotation record in corresponding OB. The first bit represents a sign of annotation record start; the second through fourth bits encode storage size of beginning (Beg) offset from the previous record; the fifth and sixth bits encode storage size of distance between End and Beg for current record; the seventh bit is the direction sign of the block offset; the eighth bit indicates the storage size of the block offset to average. (C) Workflow of random sweep. The algorithm accepts two dummy query intervals in the same chromosome (query 1 starts from 120 and ends in 150; query 2 starts from 255 and ends in 260) and efficiently executes the annotation intersection across a corresponding chromosome by leveraging the VarNote index system and original annotation database. The query 1 is first stream-compared with position information of each SROB in the VarNote index file (.vi file) to determine intersected ROBs (query 1 overlaps with ROB2 and ROB3). The algorithm directly skips unrelated ROBs and quickly locates the intersected ROBs using random access (ROB1 is completely skipped in the following searching). Because ROB only contains guery-dependent information of annotation records, VarNote can sweep the ROB more efficiently with saved disk reads (query 1 intersects an annotation record in the ROB2, and intersects another annotation record in the ROB3). Once all intersected annotation records within a ROB are identified, the algorithm can instantly seek full annotation information of record hits from the corresponding OB (only needs to seek two records in the OB2 and OB3 of annotation database for query 1). Similarly, VarNote skips over ROB4 and ROB5, only sweeps across ROB6, and finally seeks an annotation record at OB6 for query 2.

variants and annotation databases across the whole chromosome. To comprehensively evaluate VarNote in comparison with existing interval-level annotation tools, we first established a set of benchmark data sets encompassing various data scales and feature distributions (Supplemental Table S1). Six query variant data sets were generated from a genome-wide genotyping chip (A375_ chip), targeted sequencing (NA12878_Amp), whole-exome sequencing (WES) (NA12878_WES, A375_SM), whole-genome sequencing (WGS) (NA12878_WGS), and all known variants in The 1000 Genomes Project phase3 (1000G_p3); they ranged from highly unbalanced and sparse queries to highly balanced and dense queries (Fig. 2A; Supplemental Fig. S5). Several commonly used annotation databases were also prepared, including functional prediction of all potential nonsynonymous single-nucleotide variants from dbNSFP (Liu et al. 2016b), aggregated ChIP-seq peak calling results of human transcription factors from CistromeDB (Cistrome_TF) (Zheng et al. 2019), deleteriousness scores of all possible SNVs from CADD (CADD_score), and 114 related annotations (CADD_anno). These annotation databases represent distinct distributions of interspersed (dbNSFP), overlapped (Cistrome_TF), or tandem (CADD_score or CADD_anno) genomic features, respectively, which could serve as comprehensive benchmarks to evaluate the performance of VarNote and exiting algorithms (Fig. 2B; Supplemental Fig. S6).





Figure 2. Comparison of VarNote with existing tools for interval-level annotations. (*A*) The genomic distribution of six query variant data sets across Chromosome 1 of the human reference genome, including variant call result of The 1000 Genomes Project phase3 (1000G_p3), variant call result of 10x Genomics Chromium whole-genome sequencing for NA12878 (NA12878_WCS), variant call result of Nextera Rapid Capture Exome and Expanded Exome whole-exome sequencing for NA12878 (NA12878_WES), variant call result of Ion AmpliSeq Exome capture sequencing data for NA12878 (NA12878_MCS), variant call result of A375 cell line (A375_chip), and somatic mutation call result of whole-exome sequencing data for A375 cell line (A375_SM). These data sets span the highly unbalanced and sparse queries to the highly balanced and dense queries. (*B*) The genomic distribution of three annotation databases across Chromosome 1 of the human reference genome, including functional prediction and annotation of all potential nonsynonymous SNVs (dbNSFP), Cistrome aggregated ChIP-seq peak calling result of human transcription factors (Cistrome_TF), CADD deleteriousness score, and related annotation of all possible SNVs (CADD_anno). (C) The runtime comparisons among VarNote, BEDTools, BEDOPS, BCFtools, VEP, vcfanno, and GIGGLE for sequencing variants at different scales and commonly used annotation databases. (*D*) The number of runtime comparisons between VarNote and other methods within corresponding speed ratio intervals. (G) The runtime ratio distribution for processing long annotation database CADD_anno over short annotation database CADD_score.

Using the aforementioned six query data sets and four annotation databases, we constructed 24 combinatory tests for the following evaluations. We compared the performance of VarNote with five state-of-the-art bioinformatics tools for intersecting genomic features. These commonly used, representative tools can achieve interval-level overlap annotation based on distinct algorithms, including BEDTools, BEDOPS, BCFtools, VEP, GIGGLE, and vcfanno (Supplemental Table S2). Overall, VarNote runs more quickly than the previous methods and outperforms them in almost all combinatory tests in which the benchmark data sets hold various data scales and feature distributions (Supplemental Fig. S7). Specifically, VarNote shows significantly extended scalability and is well-adapted to increasing query intervals and annotation databases (Fig. 2C). In contrast, the runtime of random-access-derived methods, like VEP and GIGGLE, tightly correlates with the data scale of queries, reaching hours when annotating a WGS data set. The chromosome-sweep-derived methods (BEDTools, BEDOPS, and BCFtools) show low efficiency when annotating targeted sequencing and WES data, especially when the annotation database is huge. Although vcfanno introduces random access into the chromosome-sweep algorithm, its moderate performance largely depends on the feature distributions of the query and annotation database (Fig. 2C).

In addition to excellent scalability, VarNote shows ultrafast genomic feature intersection, being between 123 and 1514 times quicker than the other methods for the 24 benchmarks (Supplemental Fig. S7; Supplemental Table S3). VarNote finishes half of the tests within 100 sec, whereas the median runtimes of

other tools are $\sim 1000-4000$ sec for all the designed tests. The distribution of VarNote's runtimes significantly deviate from those of the other methods (Mann–Whitney U test) (Fig. 2D). VarNote runs more quickly than the other tools in 135 out of 144 (93.75%) comparisons but is slightly inferior in some evaluations involving a small annotation database and huge query. However, the second best tool, BEDOPS, only wins 69.44% of comparisons, revealing that VarNote can adapt to the full range of data scales and feature distributions while maintaining stably superior performance (Supplemental Fig. S8). Among the preceding runtimes, 65.98%, 32.64%, and 18.06% of VarNote's were more than 10, 50, and 100 times quicker, respectively, than those of the other algorithms, representing a powerful improvement (Fig. 2E,F). In addition, VarNote and GIGGLE have similar runtimes when processing the short annotation database CADD_score and the long annotation database CADD_anno, whereas the other tools show ~10 times longer runtimes, which implies that unnecessary disk reads can be largely saved by using an auxiliary positioning file (Fig. 2G).

Fast and parallelizable variant-level annotations

Efficient and accurate extraction of genomic features from annotation databases facilitates variant interpretation, particularly for the increasingly high volumes of DNA sequencing data gathered in the precision medicine era. However, current variant-level annotation tools coupled with different intersection algorithms either can barely process WGS variants or show limited efficiency for huge annotation databases. The majority of intersection algorithms do not support multithreading, which leads current variant annotation tools to implement split-and-join parallelization only by Tabix index and random access. To investigate the ability of Var-Note for all-around variant-level annotations and multithreading tasks, we selected three variant call results as query data sets (targeted sequencing, WES, and WGS) and extracted designed feature fields across three genome-wide annotation databases: dbNSFP, gnomAD (Karczewski et al. 2020), and CADD_anno. We also compared VarNote's performance against three frequently adopted and high-powered variant annotation methods, VEP, BCFtools, and vcfanno, which present relatively strong performance using random access, chromosome sweep, and mixed strategies, respectively (Supplemental Table S4).

Generally, with a single thread, VarNote outperforms the other tools, running between ~5 and 500 times more quickly in all the benchmarks. It takes <2 min to annotate NA12878 Amp targeted sequencing variants with any of the prepared annotation databases, whereas the other tools need up to hours for a large annotation database such as CADD_anno. In addition, VarNote completes common annotation tasks for WES data within 7 min. For example, it only takes 20 sec to annotate NA12878_WES variants by extracting dbNSFP nonsynonymous SNVs functional prediction scores, whereas the other tools require ~500-1700 sec. Moreover, to annotate WGS data containing several million variants, VarNote performs the most time-consuming CADD_anno task within 35 min using a single thread. In contrast, BCFtools, vcfanno, and VEP require 6, 27, and 90 h to execute the same job, respectively, making them unsuitable for personal genome applications (Fig. 3A).

We assessed multithreading performance by having selected tools perform the same evaluations using two to eight threads. Using more than four threads, VarNote can reduce the runtime of WGS annotation tasks to ~ 10 min and lower the runtime

of WES annotation tasks to nearly 1 min. When annotating targeted sequencing variants, it is more than 1000 times faster than chromosome-sweep-derived methods like BCFtools because of its restriction of parallelization (Fig. 3A). For the most time-consuming WGS annotation tasks, the speedup ratio curves of VarNote are generally sublinear and show better parallelization efficiency than other methods (Fig. 3B,C), further indicating that VarNote's index system can significantly eliminate input/output (I/O) bottlenecks for large-scale variant annotations. Because BCFtools only supports multithreading for output compression, increasing the number of cores barely improved the runtime.

Advanced functions of VarNote for versatile genomic feature intersection and variant annotation

We designed several advanced features to improve the usability of VarNote in complex annotation tasks (Fig. 4). First, because most existing annotation databases are indexed by Tabix, VarNote also provides a random-sweep searching based on the Tabix index (Supplemental Fig. S9). By performing the same tests for variantlevel annotation, we found that VarNote's Tabix mode still outperformed other tools, particularly when multithreading was applied (Supplemental Fig. S10). This implies that VarNote can faithfully process existing annotation resources without reindexing them. However, for large-scale and frequently used annotation data sets such as CADD, gnomAD, and dbNSFP, we strongly suggest using VarNote's index system to gain speed. By introducing extra file pointers, VarNote is able to support random sweep at multiple annotation databases. This is crucial to personal genome annotation tasks involving annotation resources spanning different contexts and biological domains, such as allele frequency, conservation, and functional prediction scores. In addition, VarNote allows remote annotation via FTP/HTTP, especially for big data sets that are time-consuming to download. Owing to the VarNote positioning file that only keeps query-dependent information, such remote queries differ from Tabix in having significantly reduced network data transmission load and allowing multithreading. Finally, VarNote also supports quick counting of the number of intersected features only based on the VarNote positioning file, which will be most efficient and suitable to prioritize context-dependent annotations through their relevance to the set of query variants, such as to identify the causal tissues/cell types for genome-wide association study (GWAS) variants (Huang et al. 2018) or to colocalize ChIP-seq binding events with particular transcription factors (Kanduri et al. 2019).

To extend its flexibility, VarNote allows users to describe specified operations in a configuration file to process customizable features with complex data structures (Fig. 4). For example, compared with existing tools that usually require uniformly formatted annotations such as BED (0-based) or VCF (1-based) files, VarNote can extract annotation fields from any indexed tab-delimited annotation files. This could save time in format transformation and additional disk space for large annotation databases, such as CADD and dbNSFP. When multiple annotation databases are available, VarNote allows feature extraction using both intervallevel overlap and variant-level exact matching. It also has an annotation mode supporting allele-specific variant annotation for SNV/ indel and region-specific annotation for structure variations. Selected fields of intersected outputs can be extracted and filled to the same line of the input query according to the annotation configuration file. To facilitate the integration of VarNote into

Huang et al.



Figure 3. Comparison of VarNote with existing tools for variant-level annotations. (*A*) The runtime comparisons of variant annotation using multithreading. Three variant call results as query data sets (NA12878_Amp, NA12878_WES, and NA12878_WGS) and three genome-wide annotation databases (dbNSFP, gnomAD, and CADD_anno) were used. (*B*) The parallelization efficiency comparisons for variant annotation task of NA12878_WGS querying on gnomAD. (*C*) The parallelization efficiency comparisons for variant annotation task of NA12878_WGS querying on CADD_anno.

advanced genomic programs, we also provide an application programming interface for developers.

Applications of VarNote to genome-scale prioritization of functional and pathogenic regulatory variants

GWASs have identified many genetic variants associated with hundreds of medical traits and diseases, and most of these associations are suggested to be mediated by context-specific regulatory codes in the noncoding human genome (Li et al. 2016). For Mendelian diseases and cancer, WGS technologies are frequently incorporated into the exploration of noncoding pathogenic variants from patients' entire genomes (Castel et al. 2018). However, very few computational tools have been developed to efficiently manipulate such genome-scale data and accurately prioritize the true disease-causal regulatory variants. To show the applicability and efficiency of VarNote for identifying functional and pathogenic regulatory variants at genome-scale, we integrated basewise variant annotations and several state-of-the-art regulatory variant prediction methods to develop three online computational pipelines (http://mulinlab.org/varnote/application.html): (1) disease-causal regulatory variants prioritization for GWAS results; (2) pathogenic regulatory variants prioritization for rare inherited diseases; and (3) driver regulatory variants prioritization for cancers (Fig. 5A).

Efficient and accurate prioritization of GWAS causal regulatory variants by integrating VarNote and tissue-/cell type-specific epigenomes

Although statistical fine-mapping of GWAS summary data provides a valid avenue to identify causal variants, it usually fails to narrow down the likely causal variants with extremely high linkage disequilibrium (LD) in each credible set and cannot evaluate context-dependent effects for regulatory variants in the noncoding genome (Schaid et al. 2018; Wang et al. 2020). Recently, several tissue-/cell type–specific regulatory variant prediction methods have been developed based on large-scale epigenomic features (Rojano et al. 2019), but no computational tool can leverage these methods to prioritize the potential causal regulatory variants from GWAS signals. By incorporating 127 Roadmap tissue-/cell type– specific epigenomic features, 1000 Genomes LD information, and five recent context-dependent regulatory variant prediction



Figure 4. The workflow and supported features of the VarNote annotation program. It accepts an input query (VCF, BED, or TAB format) and intersects annotation records from local or remote databases, then extracts annotation fields according to predefined annotation configuration, and finally outputs annotation results. It also supports many advanced features (shown in red).

models—cepip (Li et al. 2017b), GenoSkyline-Plus (Lu et al. 2017), FUN-LDA (Backenroth et al. 2018), GenoNet (He et al. 2018), and FitCons2 (Gulko and Siepel 2019)—we implemented a comprehensive and fast pipeline, called VarNote-REG, to accurately prioritize GWAS casual regulatory variants based on the VarNote variant annotation framework. Simplistically, given GWAS leading signals and a selected trait-matched tissue/cell type, VarNote-REG will first identify all associated variants in the defined LD block and extract epigenomic features for them. To prioritize the context-dependent regulatory potential for variants in each LD block, VarNote-REG introduces a combined rank score based on five state-of-the-art prediction scores (Methods).

VarNote-REG can complete prioritization jobs for common GWAS results, such as 161 fine-mapped loci for coronary artery disease (van der Harst and Verweij 2018) and 154 fine-mapped loci for inflammatory bowel disease (Huang et al. 2017a), within 2 min. We benchmarked the performance of VarNote-REG using PICS GWAS fine-mapped variants for 21 autoimmune diseases (Farh et al. 2015). We first filtered out LD blocks either associated with autoimmune diseases possibly driven by nonregulatory effects or containing limited highly linked (LD R² \geq 0.8) noncausal variants. For the remaining GWAS signals, we tested whether the variants with high PICS causal probability could be ranked above other highly linked variants in the matched tissue/cell type and compared the performances of various prediction methods (Methods). Using predication scores from 16 ENCODE cell types, we found that the ranking of causal variants within each causal

LD block varied with both the cell type and the prediction method. The causal variants were generally higher ranking in the E116 lymphoblastoid cell line than in other cell types (Supplemental Fig. S11), indicating that selection of relevant cell types matching autoimmune disease may improve the prioritization of true causal regulatory variants. By selecting E116 lymphoblastoid-specific prediction and partitioning the fine-mapped causal variants into three separate groups, we also observed that variants in the \geq 0.3 PICS probability group obtained higher ranks than those with low PICS probability. Particularly, our combined rank score showed better discriminatory ability than each separate method, that is, comparing the \geq 0.3 PICS probability group with the [0.05 0.3) and <0.05 PICS probability groups (Fig. SB). These results suggest that VarNote-REG could efficiently and accurately prioritize disease-causal regulatory variants.

Fast and whole-genome prioritization of pathogenic regulatory variants for rare inherited diseases

Rare noncoding variants can cause inherited disorders by affecting the function of regulatory elements, and recent genetic studies of Mendelian disease have applied WGS to identify pathogenic regulatory variants (Weedon et al. 2014; Marshall et al. 2020). Several computational methods have also been developed to prioritize the pathogenesis/deleteriousness of regulatory variants (Smedley et al. 2016). However, accurate identification of disease-causal regulatory variants from family-based WGS data requires time-



Huang et al.

Figure 5. The applied pipelines and performance evaluations of VarNote for disease-causal variant prioritization. (*A*) The function summary of three designed pipelines based on VarNote framework. (*B*) The box plot of rank ratio for each PICS causal variant among different PICS probability intervals; the rank ratio is measured by the rank of observed variant/total number of investigated variants (including extended highly linked variant in LD) in each GWAS signal. (*C*) The box plot of rank ratio for each spike-in pathogenic variant using simulated WGS variants; the rank ratio is measured by the rank of pathogenic variant using simulated individual genome, tested by one-tailed Mann–Whitney *U* test. (*D*) The box plot of rank ratio for each spike-in somatic eQTL mutation using simulated cancer genome profiles; the rank ratio is measured by the rank of spike-in somatic eQTL mutations after filtrations in each simulated cancer genome, tested by one-tailed Mann–Whitney *U* test.

consuming filtrations of genome-scale annotations as well as integration of pathogenic regulatory variant predictions. By introducing several unique filtration strategies and integrating base-wise pathogenic prediction scores, we developed a fast and accurate pipeline, VarNote-PAT, for geneticists and clinicians to prioritize likely pathogenic regulatory variants based on WGS data. Given a VCF file and matching pedigree file, VarNote-PAT can narrow down the candidate variants based on different filtrations, including variant quality, allele frequency, genetic inheritance mode, variant consequence, and tissue-/cell type–specific epigenomic features. In addition, VarNote-PAT combines seven recent prediction methods to improve the prioritization of pathogenic regulatory variants (Methods).

Building on the VarNote framework, VarNote-PAT can finish a complete prioritization task on trio-based WGS data (~5 million variants) within 60 min, whereas an existing online program, Genomiser (Smedley et al. 2016), can only deal with 100,000 variants in each run. To evaluate the performance of pathogenic variant prioritization, we first curated 18 pathogenic regulatory variants of rare inherited diseases that have been experimentally validated by different functional assays (Supplemental Table S5). We then simulated WGS data of 503 subjects by spiking each validated pathogenic variant into individual genomes from The 1000 Genomes Project EUR population (Methods). By filtering and scoring WGS variants for each simulated genome with the VarNote-PAT pipeline, we found that the ranks of spike-in pathogenic variants averaged in the top 6.5%. Compared with existing prediction scores, such as ReMM (Smedley et al. 2016) and CADD, we observed that the validated pathogenic variants were ranked significantly higher using our combined scores (Mann–Whitney *U* test) (Fig. SC). These evaluations show that VarNote-PAT can greatly improve the speed and accuracy of discovering WGS-based rare pathogenic regulatory variants.

Filtration and prioritization of regulatory somatic mutations in personal cancer genomes using VarNote

Recurrent somatic mutations in the noncoding regulatory region have been revealed as cancer drivers (Yang and Adli 2019). However, no computational pipeline has been specifically designed to screen and prioritize cancer driver regulatory mutations for a given individual cancer genome profile. Here, we encapsulated VarNote and several annotation databases to develop an online pipeline, called VarNote-CAN, for regulatory somatic mutation

prioritization. VarNote-CAN first filters germline variants and nonrecurrent mutations and then annotates recurrent somatic mutations using tissue-/cell type-specific epigenomic features. It uses our cancer driver regulatory mutation prediction model, regBase-CAN (Zhang et al. 2019), to prioritize the remaining candidate mutations. We benchmarked the performance of VarNote-CAN using 1950 simulated personal cancer genomes based on 158 highly recurrent mutations in the somatic eQTL intervals identified by a large-scale pan-cancer whole-genome analysis (PCAWG Transcriptome Core Group et al. 2020) (Methods). Compared with existing prediction scores for cancer regulatory mutations, such as FunSeq2 (Fu et al. 2014) and CScape (Rogers et al. 2017), the overall rank of somatic eQTL SNVs in all cancer genomes was higher using regBase-CAN (Mann-Whitney Utest) (Fig. 5D), indicating the applicability of VarNote-CAN to whole cancer genome analysis.

Discussion

Large-scale genomic sequencing together with constantly evolving biotechnologies in functional genomics have posed great challenges to the efficient annotation of variant functions as well as interpreting their causal links with diseases (Starita et al. 2017). An immediate challenge is the development of rapid computational methods that scale to the WGS variants and the vast amount of functional annotations across the entire genome. To address these efficiency and scalability issues, we have implemented a new index schema to tailor annotation databases. Building on this, a fast random-sweep search algorithm was designed with coherent support for multithreading. In contrast to previous strategies that merely operate original annotation databases or repetitively decompress gzip blocks, VarNote greatly saves excessive disk reads and significantly enhances scalability, thereby achieving rapid annotation for large-scale genomic features. The efficiency and scalability of VarNote make it potentially broadly applicable to large-scale genomic feature annotations. To further extend the convenience of human variant annotations for clinicians and biologists without the experience of command line operation, we designed three web tools that allow users to upload well-formatted variant lists (VCF, BED, or tab-delimited) and prioritize disease-causal regulatory variants in different scenarios. Using GWAS fine-mapped variants and simulated individual genomes, we have shown the excellent usability of the designed pipelines and demonstrated the efficient diagnosis potential afforded by VarNote's index system and ultrafast search algorithm.

Existing interval intersection and variant annotation algorithms either cannot scale to large data sets (e.g., random-accessderived methods) or lack efficiency on small and moderate queries (e.g., chromosome-sweep-derived methods). Although tools such as vcfanno use the Tabix index to implement streaming queries for sorted inputs, they usually introduce many repetitive block decompositions and unnecessary disk accesses. To the best of our knowledge, VarNote is the first tool that can adapt well to query variants and annotation databases with different levels of data scales and feature distributions. From highly unbalanced queries (e.g., variants from targeted sequencing) to extremely balanced 1000G variants, VarNote, in comparison to other existing methods, consistently shows superior performance to extract the target fields from large-scale annotations. Nevertheless, there are some rare worst-case situations in which the runtime of VarNote could slightly fall behind those of other methods; these include cases with genome features in both the query and annotation database

being largely similar or the annotation database being small while the query is very large. Our previous tests show that VarNote works best when the query variants are moderate (like WES and WGS data) and the database is huge (like CADD with full annotations), which involves using exactly the same short slabs as both randomaccess-derived methods and chromosome-sweep-derived methods. Because all of the benchmarks in this study were performed on SAS hard drives, the bottleneck of disk I/O could be reached when using a large number of threads, particularly for those tools equipped with faster searching algorithms. We suggest using SSD hard drives for large-scale annotation tasks, which will not only reduce random access time but also allow better parallelization efficiency during annotation.

Many other sophisticated computational tools could be used to annotate variants in genetic and genomic studies, for example, ANNOVAR (Wang et al. 2010), SnpEff (Cingolani et al. 2012), KGGSeq (Li et al. 2017a), WGSA (Liu et al. 2016a), Bystro (Kotlar et al. 2018), AIList (Feng et al. 2019), and Oncotator (Ramos et al. 2015). Because current variant annotation methods usually adopt similar intersection and feature extraction algorithms, or barely support customized operations, we only considered representative and relatively efficient ones in our completely fair benchmarks. Unlike many gene-based and filter-oriented annotation tools, VarNote concentrates on region-based and allele-specific annotation tasks for which many exiting methods substantially lack scalability (such as running out of memory or extremely long running times) when dealing with large-scale genomic features. In conjunction with accurate gene-based and filter-oriented annotation tools, VarNote will significantly accelerate personal genome interpretation in the precision medicine era.

Methods

The VarNote index strategy

The VarNote index comprises two main constructing steps, including generations of VarNote positioning file and VarNote index file. VarNote accepts a position-sorted (first by sequence name and then by leftmost coordinate) and block-compressed gzip (bgzip) annotation database and converts it to a new bgzip file that only keeps query-dependent information (Supplemental Fig. S1). To maximally reduce the file size and significantly save the disk accesses, VarNote transforms each original compressed block (OB) into a reduced virtual block (ROB). Specifically, for each OB that contains at most 2¹⁶ bytes, block summary information together with position information (including chromosome position and file block position) of each record are calculated and encoded to constitute ROB. The block summary information includes a unique 64-bit OB address (defined by bgzip), position information of first record, and average block offset of all records in the current OB. The position information of each record involves a record flag sign, position offset, and block offset to average. To further compress the position information of each record, an 8-bit "Record-Flag" is used to dynamically determine the exact storage volume of chromosome position offset and block offset for different records. Thus, this bit encoding strategy enables the algorithm to store consecutive base-wise chromosome positions with only 2 bytes, significantly lessening the overall storage space of ROB. In most situations, VarNote can transform each $\hat{2}^{16}$ bytes OB into $\sim 2^9 - 2^{10}$ bytes ROB, which achieves an approximately 100 times size reduction of original annotation database. Finally, a bgzip byte stream of sequential ROBs is used to generate the Var-Note positioning file (Supplemental Fig. S2A). To facilitate the

Huang et al.

efficient identification of associated ROBs intersecting with queries, VarNote creates the linear index file that contains summary information of each ROB (SROB). The SROB includes a unique 64-bit address of ROB start in the VarNote positioning file, initial beginning position of the current ROB, as well as spanning length of all records in the current ROB (Supplemental Figs. S2B, S3). The combination of the VarNote positioning file and the VarNote index file achieves fast retrieval of large numbers of query intervals.

Random-sweep searching algorithm

VarNote intellectively leverages random-access and chromosome sweep for efficient interval intersection. The same as an annotation database, query intervals/variants should be sorted first by sequence name and then by leftmost coordinate. The algorithm will first load the VarNote index and then sequentially sweep SROBs, by streaming comparison of the position information between query and each ROB, to find the first intersected ROB for initial query interval. The intersected ROB is random accessed from the VarNote positioning file through the 64-bit address of ROB start. Then the VarNote chromosome-sweep algorithm is used to decode ROB content and identify record hits. It uses a global linked list to cache position information of intersected records to make sure the sweeping is unreturned. In detail, VarNote will cache intersected records after finishing the sweeping procedure of a query interval, and then it checks whether there are any overlaps in the global linked list before it sweeps the following annotation records when the next query comes. To ensure memory-efficient searching, VarNote removes those records that no longer intersect with the following query intervals from the global linked list. In addition, VarNote uses file pointers to keep and synchronize chromosome position, and therefore makes sure the sweeping process continues. Also, the query interval may span multiple ROBs, therefore the sweeping will continue in the following ROBs until the end position of the query interval is less than the beginning position of the next ROB. Once the sweeping is finished for query intervals and the next ROB does not overlap with it, VarNote will directly seek and parse the annotation information of record hits from the corresponding OB in the annotation database (Supplemental Fig. S4). The VarNote random-sweep algorithm iteratively executes the above processes for all query intervals in a scalable manner. It also ensures that associated ROBs and OBs are decompressed only once, that is, using Intel Genomics Kernel Library (Guilford et al. 2017) for speeding up, during the whole job, which significantly save the most timeconsuming disk access and block decompression (see the pseudocode of the random-sweep searching algorithm in Supplemental Table S6).

Random-sweep searching on Tabix index

To facilitate querying when only the Tabix index is available, we also implemented a random-sweep searching algorithm based on Tabix binning and linear indexes. Generally, each chromosome is partitioned into segments spanning 128-kb intervals, and query intervals belonging to separate segments will be grouped together for subsequent searching. First, for each query interval group, we used the Tabix algorithm to identify and merge associated bins, then assemble corresponding bgzip blocks. Second, the aforementioned chromosome-sweep algorithm is used to intersect annotation records across assembled blocks, where nonconsecutive or unassociated blocks can be randomly accessed or skipped (Supplemental Fig. S9). Iteratively, the algorithm finishes all of the query interval groups.

Owing to the inherent attribute of the VarNote random-sweep algorithm, we can easily implement a parallel version of VarNote using a MapReduce programming idea. More specifically, input query file is equally partitioned to *N* subsets (*N* is a given number of threads), and for each subset VarNote independently executes random-sweep searching against annotation databases. To ensure a sorted output, VarNote merges the annotation result of all query subsets by the original order. In addition, using network streaming connection, VarNote supports remote access when huge annotation databases locate in the remote FTP/HTTP site.

Benchmark data set and environment

To comprehensively evaluate the performance of VarNote at different scales of the query variant and annotation database, we downloaded and compiled commonly used data sets from several public repositories. For the query data set, variant calling results of the NA12878 genome for AmpliSeq targeted sequencing, wholeexome sequencing (WES), and whole-genome sequencing (WGS) were downloaded from GIAB FTP (Zook et al. 2016). Variant calling result of The 1000 Genomes Project phase3 (1000G) was downloaded from EBI FTP (The 1000 Genomes Project Consortium 2015). Somatic mutation calling and germline variant genotyping results of A375 cancer cell line were downloaded from GDSC (Iorio et al. 2016). For annotation databases, we downloaded files from gnomAD (known whole-genome variant allele frequency), dbNSFP (functional prediction and annotation of all potential nonsynonymous single-nucleotide variants), CADD (deleteriousness score and related annotation of all possible single-nucleotide variants), and Cistrome Human_TF (aggregated ChIP-seq peak calling result of human transcription factors). For a data set with separate chromosome or assay files, we merged them into one annotation database (see details in Supplemental Table S1). All tests in this study were performed on a server with Intel Xeon 2.60 GHz E5-2690 v4 CPU and RAID5 mode on eight 7200 RPM SAS hard drives.

Evaluation for interval-level annotations

We compared the performance of VarNote intersection function with several well-performed tools in each algorithm category, including BEDTools, BEDOPS, BCFtools, VEP, vcfanno, and GIGGLE. Using prepared VCF inputs (AmpliSeq, WES and WGS, 1000G) and BED databases (dbNSFP, Cistrome Human_TF, and CADD with or without related annotations) at different data sizes, we tested the runtime and scalability of each software for intervallevel annotations. Because BEDTools, BEDOPS, BCFtools, and GIGGLE do not support multiple threads execution, all intervallevel comparisons were based on single thread mode (see details in Supplemental Table S2). For VEP, the actual runtime of the intersection was calculated by subtracting the runtime of the gene-based annotation.

Evaluation for variant-level annotations

We evaluated the performance for VarNote and three representative variant annotation tools (BCFtools, VEP, and vcfanno), which use unique search algorithms, respectively. Three VCF queries (NA12878 AmpliSeq, WES, and WGS) were used to test the allele-specific annotations against one VCF database (gnomAD) and two BED databases (dbNSFP and CADD with full annotations). Besides, we also inspected the performance of these programs at multiple threads (see details in Supplemental Table S4).

Implementation of the VarNote-REG pipeline

Genotype data of different populations were retrieved from The 1000 Genomes Project phase 3 release, and LD was estimated by a correlation method. Consolidated and imputed epigenomes from 127 human tissues/cell lines were downloaded from the web portal of the NIH Roadmap Epigenomics project (Roadmap Epigenomics Consortium et al. 2015), which includes ChIP-seq narrow peaks for eight histone modifications (H3K4me1, H3K4me2, H3K4me3, H3K27ac, H3K27me3, H3K36me3, H3K79me2, and H3K9me3) and DNase-seq peaks. Jannovar (Jäger et al. 2014) was used to map variant consequences. Five tissue-/cell type-specific regulatory variants prediction methods-cepip (Li et al. 2017b), GenoSkyline-Plus (Lu et al. 2017), FUN-LDA (Backenroth et al. 2018), GenoNet (He et al. 2018), and FitCons2 (Gulko and Siepel 2019)-were used to prioritize likely causal variants in each GWAS signal. For each candidate variant in its LD block, the ranks of prediction scores were combined by calculating rank product. By encapsulating VarNote programming interfaces, the VarNote-REG pipeline introduces three main steps to prioritize regulatory variant in each input GWAS signal: (1) variant normalization and LD expansion; (2) tissue-/cell type-specific epigenomic features annotation; and (3) tissue-/cell type-specific regulatory potential prioritization. The online VarNote-REG pipeline and visualization is available at http://mulinlab.org/varnote/application.html#REG.

Evaluation of VarNote-REG for tissue-/cell type-specific regulatory variant prioritization

Candidate causal variants for 21 autoimmune diseases were downloaded from a PICS GWAS fine-mapping study (Farh et al. 2015). We first used Jannovar to annotate the variant consequence and excluded the signals (leading variant-associated LD block) in which the causal variant with the largest PICS probability having protein-coding or splicing-altering consequences. For each leading variant-associated GWAS signal, we extended the linked variants in this signal $(\mathbb{R}^2 \ge 0.8)$ and treated them as noncausal variants. We removed the GWAS signal once the extended noncausal variants are insufficient (less than five variants). We used VarNote to retrieve tissue-/cell type-specific regulatory variant prediction scores on 16 ENCODE cell types for five existing methods and calculated combined scores by rank product. We then tested whether the causal variants could be ranked higher than highly linked noncausal variants in different tissues/cell types and compared the performance among used prediction methods.

Implementation of the VarNote-PAT pipeline

Allele frequency information for different populations were downloaded from The 1000 Genomes Project phase 3 release and gnomAD v2.1 (Karczewski et al. 2020). Four genetic inheritance modes were supported in VarNote-PAT, including autosomal dominant/recessive and X-linked dominant/recessive inheritances. Variant filtration strategies were similar to our previous WES platform wKGGSeq (Li et al. 2015). We specially introduced tissue/celltype-specific epigenomic feature annotation and filtration using 127 Roadmap epigenomic profiles. We combined the prediction scores of seven methods-CADD v1.4 (Kircher et al. 2014), ReMM (Smedley et al. 2016), Eigen (Ionita-Laza et al. 2016), GenoCanyon (Lu et al. 2015), FATHMM_MKL (Shihab et al. 2015), FATHMM-XF (Rogers et al. 2018), and LINSIGHT (Huang et al. 2017b)-for evaluating pathogenic noncoding variant by a rank product method. Overall, the VarNote-PAT pipeline incorporated four main steps to rank pathogenic regulatory variant from WGS variants: (1) filtration on allele frequency; (2) filtration on genetic inheritance mode; (3) filtration on variant consequences and epigenomic features; and (4) pathogenicity prioritization. The online VarNote-PAT pipeline and visualization is available at http:// mulinlab.org/varnote/application.html#PAT.

Evaluation of VarNote-PAT for rare pathogenic variant prioritization

Known rare pathogenic regulatory variants (minor allele frequency in EUR < 0.001) were collected from ClinVar (Landrum et al. 2020), Genomiser (Smedley et al. 2016), RegBase (Zhang et al. 2019), NCBoost (Caron et al. 2019), and CDTS (di Iulio et al. 2018). We used Jannovar to annotate the variant consequence and excluded the variants with protein-coding or splicing-altering attributes. Because some of these pathogenic variants were not confirmed by functional study, we manually inspected which pathogenic variants were validated by functional experiments (such as luciferase reporter assay of promoter/enhancer, electrophoretic mobility shift assay, chromatin conformation capture assay, etc.) from the original publication and other literature. We simulated WGS results of pathogenic variant carrier by spiking each validated variant into each individual genome (503 individuals in EUR population from The 1000 Genomes Project) according to dominant disease inheritance mode. We then tested whether the spike-in pathogenic variants could be top-ranked in the simulated individual genomes (filtering criteria: allele frequency: less than 0.001 in EUR; variant consequence: protein-coding and splicing-altering excluded; genome region: 10 kb upstream and downstream from the gene promoter) and compared the performance among our combined strategy and existing methods of pathogenic regulatory variant prediction.

Implementation of the VarNote-CAN pipeline

The used variant annotations for allele frequency and tissue/celltype-specific epigenomic features are the same as those in the VarNote-PAT pipeline. Noncoding somatic mutations and their recurrence information were downloaded from COSMIC v90 (Tate et al. 2019). We used our recent regBase-CAN score (Zhang et al. 2019) to prioritize cancer driver regulatory variants. The VarNote-CAN pipeline incorporated four main steps to rank cancer driver regulatory variants for personal cancer genome: (1) filtration on germline variants; (2) filtration on variant consequences and epigenomic features; (3) filtration on somatic recurrence; and (4) cancer driver mutation prioritization. The online VarNote-CAN pipeline and visualization is available at http://mulinlab.org/ varnote/application.html#CAN.

Evaluation of VarNote-CAN for regulatory somatic mutation prioritization

Somatic eGene and associated eQTL intervals were downloaded from a large-scale pan-cancer whole-genome analysis (PCAWG Transcriptome Core Group et al. 2020). In each eQTL interval, we identified the highly recurrent somatic mutations based on COSMIC v90 data and constructed a data set of candidate driver regulatory mutations. Because some intervals contain multiple best somatic mutations with the same recurrence, we only selected one representative by requiring a median pathogenic score on regBase-CAN. We simulated individual cancer mutation profiles by spiking the candidate driver regulatory mutation into somatic mutation calling results of each WGS cancer genome (1950 patients collected from ICGC). We then executed VarNote-CAN pipeline for all simulated personal cancer genomes (filtering criteria: variant consequence: protein-coding and splicing-altering excluded; genome region: 10 kb upstream and downstream from the gene promoter) and compared the ranks of the spike-in cancer

Huang et al.

driver mutations among our regBase-CAN score and existing somatic regulatory mutation prediction methods, including FunSeq2 v2.1.6 (Fu et al. 2014), CScape (Rogers et al. 2017), CADD v1.4 (Kircher et al. 2014), Eigen (Ionita-Laza et al. 2016), GenoCanyon (Lu et al. 2015), and FATHMM_MKL (Shihab et al. 2015).

Software availability

All source code and scripts for methods evaluation and manuscript results are available at GitHub (https://github.com/mulinlab/ VarNote) and as Supplemental Code. Software, documentation, and VarNote online applications are available at http://mulinlab .org/varnote. VarNote web servers are running at qual-core mode and SAS hard disks for genomic feature intersection and variant annotation.

Competing interest statement

The authors declare no competing interests.

Acknowledgments

This work was supported by grants from the National Natural Science Foundation of China 31871327 (M.J.L.), and Natural Science Foundation of Tianjin City 19JCJQJC63600, 18JCZDJ C34700 (M.J.L.). We thank all tool and resource providers.

Author contributions: D.H. and M.J.L. implemented the algorithms. D.H., X.Y., and M.J.L. conducted the computational analysis described in the paper. D.H. and Y.Z. developed the database and web portal. M.J.L. conceived the study and wrote the paper. H.Y., H.X., JH.W., S.Z., W.N., and P.W. evaluated the computational tool, tested the web portal, and reviewed the manuscript. L.S., C.X., M.L., JW.W., W.L., and H.S.K. suggested the tool functions and reviewed the manuscript. P.C.S. and K.W. contributed to coordination and supervision of computational methodologies and reviewed the manuscript. All authors read and approved the final manuscript.

References

- The 1000 Genomes Project Consortium. 2015. A global reference for human genetic variation. *Nature* **526:** 68–74. doi:10.1038/nature15393
- Backenroth D, He Z, Kiryluk K, Boeva V, Pethukova L, Khurana E, Christiano A, Buxbaum JD, Ionita-Laza I. 2018. FUN-LDA: a latent Dirichlet allocation model for predicting tissue-specific functional effects of noncoding variation: methods and applications. Am J Hum Genet 102: 920–942. doi:10.1016/j.ajhg.2018.03.026
- Brody JA, Morrison AC, Bis JC, O'Connell JR, Brown MR, Huffman JE, Ames DC, Carroll A, Conomos MP, Gabriel S, et al. 2017. Analysis commons, a team approach to discovery in a big-data environment for genetic epidemiology. *Nat Genet* **49**: 1560–1563. doi:10.1038/ng.3968
- Bujold D, Morais DAL, Gauthier C, Côté C, Caron M, Kwan T, Chen KC, Laperle J, Markovits AN, Pastinen T, et al. 2016. The International Human Epigenome Consortium data portal. *Cell Syst* 3: 496–499.e2. doi:10.1016/j.cels.2016.10.019
- Bycroft C, Freeman C, Petkova D, Band G, Elliott LT, Sharp K, Motyer A, Vukcevic D, Delaneau O, O'Connell J, et al. 2018. The UK Biobank resource with deep phenotyping and genomic data. *Nature* **562**: 203– 209. doi:10.1038/s41586-018-0579-z
- Caron B, Luo Y, Rausell A. 2019. NCBoost classifies pathogenic non-coding variants in mendelian diseases through supervised learning on purifying selection signals in humans. *Genome Biol* 20: 32. doi:10.1186/ s13059-019-1634-2
- Castel SE, Cervera A, Mohammadi P, Aguet F, Reverter F, Wolman A, Guigo R, Iossifov I, Vasileva A, Lappalainen T. 2018. Modified penetrance of coding variants by *cis*-regulatory variation contributes to disease risk. *Nat Genet* **50**: 1327–1334. doi:10.1038/s41588-018-0192-y
- Cingolani P, Platts A, Wang le L, Coon M, Nguyen T, Wang L, Land SJ, Lu X, Ruden DM. 2012. A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of

- Davis CA, Hitz BC, Sloan CA, Chan ET, Davidson JM, Gabdank I, Hilton JA, Jain K, Baymuradov UK, Narayanan AK, et al. 2018. The Encyclopedia of DNA elements (ENCODE): data portal update. Nucleic Acids Res 46: D794–D801. doi:10.1093/nar/gkx1081
- di Iulio J, Bartha I, Wong EHM, Yu HC, Lavrenko V, Yang D, Jung I, Hicks MA, Shah N, Kirkness EF, et al. 2018. The human noncoding genome defined by genetic diversity. *Nat Genet* **50**: 333–337. doi:10.1038/ s41588-018-0062-7
- The ENCODE Project Consortium. 2012. An integrated encyclopedia of DNA elements in the human genome. *Nature* **489:** 57–74. doi:10 .1038/nature11247
- Farh KK, Marson A, Zhu J, Kleinewietfeld M, Housley WJ, Beik S, Shoresh N, Whitton H, Ryan RJ, Shishkin AA, et al. 2015. Genetic and epigenetic fine mapping of causal autoimmune disease variants. *Nature* **518**: 337–343. doi:10.1038/nature13835
- Feng J, Ratan A, Sheffield NC. 2019. Augmented interval list: a novel data structure for efficient genomic interval search. *Bioinformatics* 35: 4907–4911. doi:10.1093/bioinformatics/btz407
- Fu Y, Liu Z, Lou S, Bedford J, Mu XJ, Yip KY, Khurana E, Gerstein M. 2014. FunSeq2: a framework for prioritizing noncoding regulatory variants in cancer. *Genome Biol* **15**: 480. doi:10.1186/s13059-014-0480-5
- Guilford J, Bergelson L, Powley G, Tucker G, Lichtenstein L, Vaidya P, Roazen D. 2017. Accelerating the compression and decompression of genomics data using GKL provided by Intel. https://www.intel.com/ content/dam/www/public/us/en/documents/white-papers/acceleratinggenomics-data-gkl-white-paper.pdf.
- Gulko B, Siepel A. 2019. An evolutionary framework for measuring epigenomic information and estimating cell-type-specific fitness consequences. *Nat Genet* **51**: 335–342. doi:10.1038/s41588-018-0300-z
- He Z, Liu L, Wang K, Ionita-Laza I. 2018. A semi-supervised approach for predicting cell-type specific functional consequences of non-coding variation using MPRAs. *Nat Commun* 9: 5199. doi:10.1038/s41467-018-07349-w
- Huang H, Fang M, Jostins L, Umićević Mirkov M, Boucher G, Anderson CA, Andersen V, Cleynen I, Cortes A, Crins F, et al. 2017a. Fine-mapping inflammatory bowel disease loci to single-variant resolution. *Nature* 547: 173–178. doi:10.1038/nature22969
- Huang YF, Gulko B, Siepel A. 2017b. Fast, scalable prediction of deleterious noncoding variants from functional and population genomic data. *Nat Genet* 49: 618–624. doi:10.1038/ng.3810
- Huang D, Yi X, Zhang S, Zheng Z, Wang P, Xuan C, Sham PC, Wang J, Li MJ. 2018. GWAS4D: multidimensional analysis of context-specific regulatory variant for human complex diseases and traits. *Nucleic Acids Res* 46: W114–W120. doi:10.1093/nar/gky407
- Ionita-Laza I, McCallum K, Xu B, Buxbaum JD. 2016. A spectral approach integrating functional genomic annotations for coding and noncoding variants. *Nat Genet* 48: 214–220. doi:10.1038/ng.3477
- Iorio F, Knijnenburg TA, Vis DJ, Bignell GR, Menden MP, Schubert M, Aben N, Goncalves E, Barthorpe S, Lightfoot H, et al. 2016. A landscape of pharmacogenomic interactions in cancer. *Cell* **166**: 740–754. doi:10 .1016/j.cell.2016.06.017
- Jäger M, Wang K, Bauer S, Smedley D, Krawitz P, Robinson PN. 2014. Jannovar: a Java library for exome annotation. *Hum Mutat* 35: 548– 555. doi:10.1002/humu.22531
- Kanduri C, Bock C, Gundersen S, Hovig E, Sandve GK. 2019. Colocalization analyses of genomic elements: approaches, recommendations and challenges. *Bioinformatics* **35**: 1615–1624. doi:10.1093/bioinformatics/ bty835
- Karczewski KJ, Francioli LC, Tiao G, Cummings BB, Alföldi J, Wang Q, Collins RL, Laricchia KM, Ganna A, Birnbaum DP, et al. 2020. The mutational constraint spectrum quantified from variation in 141,456 humans. *Nature* 581: 434–443. doi:10.1038/s41586-020-2308-7
- Kent WJ, Sugnet CW, Furey TS, Roskin KM, Pringle TH, Zahler AM, Haussler D. 2002. The human genome browser at UCSC. *Genome Res* 12: 996– 1006. doi:10.1101/gr.229102
- Kircher M, Witten DM, Jain P, O'Roak BJ, Cooper GM, Shendure J. 2014. A general framework for estimating the relative pathogenicity of human genetic variants. *Nat Genet* **46**: 310–315. doi:10.1038/ng.2892
- Kotlar AV, Trevino CE, Zwick ME, Cutler DJ, Wingo TS. 2018. Bystro: rapid online variant annotation and natural-language filtering at whole-genome scale. *Genome Biol* **19**: 14. doi:10.1186/s13059-018-1387-3
- Landrum MJ, Chitipiralla S, Brown GR, Chen C, Gu B, Hart J, Hoffman D, Jang W, Kaur K, Liu C, et al. 2020. ClinVar: improvements to accessing data. Nucleic Acids Res 48: D835–D844. doi:10.1093/nar/gkz972
- Layer RM, Pedersen BS, DiSera T, Marth GT, Gertz J, Quinlan AR. 2018. GIGGLE: a search engine for large-scale integrated genome analysis. *Nat Methods* **15:** 123–126. doi:10.1038/nmeth.4556
- Li H. 2011a. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from

sequencing data. *Bioinformatics* 27: 2987–2993. doi:10.1093/bioinformatics/btr509

- Li H. 2011b. Tabix: fast retrieval of sequence features from generic TAB-delimited files. *Bioinformatics* 27: 718–719. doi:10.1093/bioinformatics/ btq671
- Li MJ, Deng J, Wang P, Yang W, Ho SL, Sham PC, Wang J, Li M. 2015. wKGGSeq: a comprehensive strategy-based and disease-targeted online framework to facilitate exome sequencing studies of inherited disorders. *Hum Mutat* **36**: 496–503. doi:10.1002/humu.22766
- Li MJ, Liu Z, Wang P, Wong MP, Nelson MR, Kocher JP, Yeager M, Sham PC, Chanock SJ, Xia Z, et al. 2016. GWASdb v2: an update database for human genetic variants identified by genome-wide association studies. *Nucleic Acids Res* **44:** D869–D876. doi:10.1093/nar/gkv1317
- Li M, Li J, Li MJ, Pan Z, Hsu JS, Liu DJ, Zhan X, Wang J, Song Y, Sham PC. 2017a. Robust and rapid algorithms facilitate large-scale whole genome sequencing downstream analysis in an integrative framework. *Nucleic Acids Res* **45**: e75. doi:10.1093/nar/gkx019
- Li MJ, Li M, Liu Z, Yan B, Pan Z, Huang D, Liang Q, Ying D, Xu F, Yao H, et al. 2017b. cepip: context-dependent epigenomic weighting for prioritization of regulatory variants and disease-associated genes. *Genome Biol* 18: 52. doi:10.1186/s13059-017-1177-3
- Liu X, White S, Peng B, Johnson AD, Brody JA, Li AH, Huang Z, Carroll A, Wei P, Gibbs R, et al. 2016a. WGSA: an annotation pipeline for human genome sequencing studies. J Med Genet 53: 111–112. doi:10.1136/ imedgenet-2015-103423
- Liu X, Wu C, Li C, Boerwinkle E. 2016b. dbNSFP v3.0: a one-stop database of functional predictions and annotations for human nonsynonymous and splice-site SNVs. *Hum Mutat* **37**: 235–241. doi:10.1002/humu .22932
- Lu Q, Hu Y, Sun J, Cheng Y, Cheung KH, Zhao H. 2015. A statistical framework to predict functional non-coding regions in the human genome through integrated analysis of annotation data. *Sci Rep* **5**: 10576. doi:10.1038/srep10576
- Lu Q, Powles RL, Abdallah S, Ou D, Wang Q, Hu Y, Lu Y, Liu W, Li B, Mukherjee S, et al. 2017. Systematic tissue-specific functional annotation of the human genome highlights immune-related DNA elements for late-onset Alzheimer's disease. *PLoS Genet* 13: e1006933. doi:10 .1371/journal.pgen.1006933
- MacArthur DG, Manolio TA, Dimmock DP, Rehm HL, Shendure J, Abecasis GR, Adams DR, Altman RB, Antonarakis SE, Ashley EA, et al. 2014. Guidelines for investigating causality of sequence variants in human disease. *Nature* **508**: 469–476. doi:10.1038/nature13127
- Marshall CR, Bick D, Belmont JW, Taylor SL, Ashley E, Dimmock D, Jobanputra V, Kearney HM, Kulkarni S, Rehm H, et al. 2020. The medical genome initiative: moving whole-genome sequencing for rare disease diagnosis to the clinic. *Genome Med* **12**: 48. doi:10.1186/s13073-020-00748-z
- McLaren W, Gil L, Hunt SE, Riat HS, Ritchie GR, Thormann A, Flicek P, Cunningham F. 2016. The Ensembl Variant Effect Predictor. *Genome Biol* **17**: 122. doi:10.1186/s13059-016-0974-4
- Neph S, Kuehn MS, Reynolds AP, Haugen E, Thurman RE, Johnson AK, Rynes E, Maurano MT, Vierstra J, Thomas S, et al. 2012. BEDOPS: high-performance genomic feature operations. *Bioinformatics* **28**: 1919–1920. doi:10.1093/bioinformatics/bts277
- PCAWG Transcriptome Core Group, Calabrese C, Davidson NR, Demircioğlu D, Fonseca NA, He Y, Kahles A, Lehmann KV, Liu F, Shiraishi Y, et al. 2020. Genomic basis for RNA alterations in cancer. *Nature* **578**: 129–136. doi:10.1038/s41586-020-1970-0
- Pedersen BS, Layer RM, Quinlan AR. 2016. *Vcfanno*: fast, flexible annotation of genetic variants. *Genome Biol* **17**: 118. doi:10.1186/s13059-016-0973-5
- Quinlan AR, Hall IM. 2010. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* 26: 841–842. doi:10.1093/bioinfor matics/btq033
- Ramos AH, Lichtenstein L, Gupta M, Lawrence MS, Pugh TJ, Saksena G, Meyerson M, Getz G. 2015. Oncotator: cancer variant annotation tool. *Hum Mutat* 36: E2423–E2429. doi:10.1002/humu.22771

- The Roadmap Epigenomics Consortium, Kundaje A, Meuleman W, Ernst J, Bilenky M, Yen A, Heravi-Moussavi A, Kheradpour P, Zhang Z, Wang J, et al. 2015. Integrative analysis of 111 reference human epigenomes. *Nature* **518**: 317–330. doi:10.1038/nature14248
- Rogers MF, Shihab HA, Gaunt TR, Campbell C. 2017. CScape: a tool for predicting oncogenic single-point mutations in the cancer genome. *Sci Rep* 7: 11597. doi:10.1038/s41598-017-11746-4
- Rogers MF, Shihab HA, Mort M, Cooper DN, Gaunt TR, Campbell C. 2018. FATHMM-XF: accurate prediction of pathogenic point mutations via extended features. *Bioinformatics* **34**: 511–513. doi:10.1093/bioinfor matics/btx536
- Rojano E, Seoane P, Ranea JAG, Perkins JR. 2019. Regulatory variants: from detection to predicting impact. *Brief Bioinform* **20**: 1639–1654. doi:10 .1093/bib/bby039
- Schaid DJ, Chen W, Larson NB. 2018. From genome-wide associations to candidate causal variants by statistical fine-mapping. *Nat Rev Genet* 19: 491–504. doi:10.1038/s41576-018-0016-z
- Shihab HA, Rogers MF, Gough J, Mort M, Cooper DN, Day IN, Gaunt TR, Campbell C. 2015. An integrative approach to predicting the functional effects of non-coding and coding sequence variation. *Bioinformatics* 31: 1536–1543. doi:10.1093/bioinformatics/btv009
- Smedley D, Schubach M, Jacobsen JOB, Kohler S, Zemojtel T, Spielmann M, Jager M, Hochheiser H, Washington NL, McMurry JA, et al. 2016. A whole-genome analysis framework for effective identification of pathogenic regulatory variants in mendelian disease. *Am J Hum Genet* **99**: 595– 606. doi:10.1016/j.ajhg.2016.07.005
- Starita LM, Ahituv N, Dunham MJ, Kitzman JO, Roth FP, Seelig G, Shendure J, Fowler DM. 2017. Variant interpretation: functional assays to the rescue. Am J Hum Genet 101: 315–325. doi:10.1016/j.ajhg.2017.07.014
- Tate JG, Bamford S, Jubb HC, Sondka Z, Beare DM, Bindal N, Boutselakis H, Cole CG, Creatore C, Dawson E, et al. 2019. COSMIC: the catalogue of somatic mutations in cancer. *Nucleic Acids Res* **47:** D941–D947. doi:10 .1093/nar/gky1015
- van der Harst P, Verweij N. 2018. Identification of 64 novel genetic loci provides an expanded view on the genetic architecture of coronary artery disease. Circ Res 122: 433–443. doi:10.1161/CIRCRESAHA.117.312086
- Wang K, Li M, Hakonarson H. 2010. ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. *Nucleic Acids Res* 38: e164. doi:10.1093/nar/gkq603
- Wang J, Huang D, Zhou Y, Yao H, Liu H, Zhai S, Wu C, Zheng Z, Zhao K, Wang Z, et al. 2020. CAUSALdb: a database for disease/trait causal variants identified using summary statistics of genome-wide association studies. Nucleic Acids Res 48: D807–D816. doi:10.1093/nar/gkaa807
- Weedon MN, Cebola I, Patch AM, Flanagan SE, De Franco E, Caswell R, Rodríguez-Segui SA, Shaw-Smith C, Cho CH, Allen HL, et al. 2014. Recessive mutations in a distal *PTF1A* enhancer cause isolated pancreatic agenesis. *Nat Genet* **46**: 61–64. doi:10.1038/ng.2826
- Yang J, Adli M. 2019. Mapping and making sense of noncoding mutations in the genome. *Cancer Res* **79:** 4309–4314. doi:10.1158/0008-5472 .CAN-19-0905
- Zhang S, He Y, Liu H, Zhai H, Huang D, Yi X, Dong X, Wang Z, Zhao K, Zhou Y, et al. 2019. regBase: whole genome base-wise aggregation and functional prediction for human non-coding regulatory variants. *Nucleic Acids Res* 47: e134. doi:10.1093/nar/gkz774
- Zheng R, Wan C, Mei S, Qin Q, Wu Q, Sun H, Chen CH, Brown M, Zhang X, Meyer CA, et al. 2019. Cistrome Data Browser: expanded datasets and new tools for gene regulatory analysis. *Nucleic Acids Res* 47: D729– D735. doi:10.1093/nar/gky1094
- Zook JM, Catoe D, McDaniel J, Vang L, Spies N, Sidow A, Weng Z, Liu Y, Mason CE, Alexander N, et al. 2016. Extensive sequencing of seven human genomes to characterize benchmark reference materials. *Sci Data* 3: 160025. doi:10.1038/sdata.2016.25

Received June 28, 2020; accepted in revised form September 22, 2020.



Ultrafast and scalable variant annotation and prioritization with big functional genomics data

Dandan Huang, Xianfu Yi, Yao Zhou, et al.

Genome Res. 2020 30: 1789-1801 originally published online October 15, 2020 Access the most recent version at doi:10.1101/gr.267997.120

Supplemental Material	http://genome.cshlp.org/content/suppl/2020/11/17/gr.267997.120.DC1
References	This article cites 64 articles, 4 of which can be accessed free at: http://genome.cshlp.org/content/30/12/1789.full.html#ref-list-1
Creative Commons License	This article is distributed exclusively by Cold Spring Harbor Laboratory Press for the first six months after the full-issue publication date (see http://genome.cshlp.org/site/misc/terms.xhtml). After six months, it is available under a Creative Commons License (Attribution-NonCommercial 4.0 International), as described at http://creativecommons.org/licenses/by-nc/4.0/.
Email Alerting Service	Receive free email alerts when new articles cite this article - sign up in the box at the top right corner of the article or click here .





To subscribe to Genome Research go to: http://genome.cshlp.org/subscriptions

Supplemental Material

Ultrafast and scalable variant annotation and prioritization with big functional genomics data

This file includes: Supplemental Figures S1-S11, Supplemental Tables S1-S6 and the description of Supplemental Codes.

Supplemental Figures

Figure S1. Architecture of VarNote index system	3
Figure S2. The construction of VarNote positioning file and index file	4
Figure S3. The VarNote index format	5
Figure S4. Process of chromosome sweeping	6
Figure S5. The genomic distribution of six query variant datasets	7
Figure S6. The genomic distribution of three annotation databases	8
Figure S7. The runtime comparisons of interval-level annotations among six methods	9
Figure S8. The runtime comparisons between VarNote and other tools	10
Figure S9. Random-sweep searching algorithm based on Tabix index	10
Figure S10. The runtime comparisons of variant annotation using multi-threading	11
Figure S11. The boxplot of rank ratio for PICS causal variant across conditions	12



Supplemental Figure S1. Architecture of VarNote index system. Bgzip-compressed annotation database (.bgz file) will be firstly converted to VarNote positioning file (.vanno file). The system tailors and encodes information of each original compressed block in the annotation database (OB) to generate reduced virtual block that only keeps query-dependent information (ROB). The bgzip-compressed VarNote positioning file contains concatenate compressed block that stores ROB bytes (VB). Then, summary information of reduced virtual block (SROB) is linearly indexed to generate VarNote index file (.vi file).



Supplemental Figure S2. The construction of VarNote positioning file and index file. (A) ROB transformation. A normal OB usually stores 2^{16} bytes and may contains hundreds to thousands annotation records ($2^{6} \sim 2^{10}$ bytes each). Since the most space-consuming information is annotation metadata that is useless for interval query, the system only need to remember query-dependent information, called ROB. ROB comprises a block summary (11~17 bytes in total, including a unique 64-bit bgzip OB address, position information of first record and average block offset of all records in the current OB) and hundreds to thousands encoded annotation records (2~13 bytes each, including a record flag, position offset and block offset to average). (B) Linear index of ROB. The system indexes the VarNote positioning file by a series of ROB summary information (16 bytes each, including a unique 64-bit bgzip address of ROB start, initial beginning position of ROB and spanning length of all records in ROB).

Fiel	d		Description	Туре
Blo	ck Start Flag		 -1 means block start and first record distance is 1; -2 means block start and first record distance is a unsigned byte value; -3 means block start and first record distance is a unsigned short value; -4 means block start and first record distance is a int value. 	byte
	First Record in	n Block		
	Block Addres	iS	64-bit bgzip address of record in database file.	long
	Record Dista	nce	Record Distance (End minus Beg), optional (If record distance is greater than one, 1,2 or 4 bytes are allocated to store the unsigned byte, unsigned short and int distance value respectively).	0, unsigned byte, unsigned short, int
	Average Offs	et	Average block offset of all records in this block.	short, int
	Other Records	s in Block		
	RecordFlag			byte
	Record	Sign	The 1st bit represents the start of a record flag.	bit[0]
	Beg Dis	stance Sign	The 2nd-4th bits encodes the storage size of distance between the current beg and previous (additional bytes are allocated to store the difference).	bit[1-3]
	Record	Distance Sign	The 5th-6th bits encodes the storage size of record distance (additional bytes are allocated to store the difference).	bit[4-5]
	Directio	n Sign	The 7th bits encodes direction sign indicates whether the difference of block offset to average is a positive or negative value.	bit[6]
	Storage Size Sign		The 8th bits encodes a sign indicates the storage size of block offset to average (additional bytes are allocated to store the difference).	bit[7]
	Beg Distance)	Optional, if the distance between current beg and previous is greater than four, 1,2 or 4 bytes are allocated to store the unsigned byte, unsigned short and int distance value respectively.	0, unsigned byte, unsigned short, int
	Record Dista	nce	Optional, if record distance is greater than one, 1,2 or 4 bytes are allocated to store the unsigned byte, unsigned short and int distance value respectively.	0, unsigned byte, unsigned short, int
	Difference be Offset to Ave	etween Block rage Offset	Additional 1-4 bytes are allocated to store the difference, allocating 1,2 or 4 for unsigned byte, short, int value of difference respectively.	unsigned byte, short, int
Fiel	d		Description	Туре
Blo	ck Count		Total block in current chromosome	int
	Summary of E	ach Block		
	Block Addres	s	Address of this block in the VarNote file.	long
	Start Position	1	The start position (Beg) of the first record in this block.	int
	Block Distance	се	int	

Supplemental Figure S3. The VarNote index format (including .vanno and .vi files). The original database file should be bgzip-compressed.



Supplemental Figure S4. Process of chromosome sweeping. Suppose there are four query intervals (Q1-Q4, some of them overlap each other) and seven annotation records in the database (D1-D7, some of them overlap each other), VarNote sequentially intersects Q1 with annotation records to identify all record hits (Hit D1, D2, D3, D4, D5). A global linked list is used to cache these intersected records in case next query interval encounters. Then, VarNote processes Q2 by firstly inspecting whether there are any overlaps in the global linked list (Hit D1, D3, D4, D5), and then dropping useless cached records from the global linked list (Del D2). The algorithm continues to sweep and cache the following annotation records for Q2 (no hit). Iteratively, VarNote accurately finds all intersections for Q3 (hit D3, D5, D6), deletes never used records from the global linked list (Del D1, D4) until the last query interval ends.



Supplemental Figure S5. The genomic distribution of six query variant datasets across the human reference genome, including variant call result of 1000 Genomes project phase3 (1000G_p3), variant call result of 10X Genomics Chromium whole genome sequencing for NA12878 (NA12878_WGS), variant call result of Nextera Rapid Capture Exome and Expanded Exome whole exome sequencing for NA12878 (NA12878_WES), variant call result of Ion Ampliseq Exome capture sequencing data for NA12878 (NA12878_Amp), genotype call result of Affymetrix Genome-Wide Human SNP Array 6.0 data for A375 cell line (A375_chip) and somatic mutation call result of whole exome sequencing data for A375 cell line (A375_SM). These datasets span the highly unbalanced and sparse queries to the highly balanced and dense queries.



Supplemental Figure S6. The genomic distribution of three annotation databases across human reference genome, including functional prediction and annotation of all potential non-synonymous SNVs (dbNSFP), Cistrome aggregated ChIP-seq peak calling result of human transcription factors (Cistrome_TF), CADD deleteriousness score and related annotation of all possible SNVs (CADD_anno).

1000G_p3 : CADD_anno	1.4E5	2.2E4	2.3E4	6.2E5	6.2E5	1.5E5	1.1E4
NA12878_WGS : CADD_anno -	7.6E4	2.2E4	2.6E4	3.3E5	1.4E4	9.8E4	2E3
NA12878_WES : CADD_anno -	7.5E4	2.2E4	2.3E4	3E4	1.2E3	1E5	352
A375_chip : CADD_anno -	7.4E4	2E4	2.2E4	5.7E4	2.5E3	1E5	599
NA12878_Amp : CADD_anno -	7.5E4	2.2E4	2.3E4	5.6E3	151	2.8E4	50
A375_SM : CADD_anno -	8E4	2.1E4	2.3E4	4.3E3	271	3.8E4	76
1000G_p3 : CADD_score -	1.3E4	3.3E3	3.2E3	5.4E4	3.1E5	3.3E4	2.5E3
NA12878_WGS : CADD_score	6.2E3	3.3E3	3.5E3	2.9E4	8E3	1.5E4	1E3
NA12878_WES : CADD_score	6E3	3.2E3	2.9E3	2.4E3	695	1E4	142
A375_chip : CADD_score -	6.2E3	3E3	3.2E3	5.4E3	1.6E3	1.2E4	147
NA12878_Amp : CADD_score -	6.4E3	3.2E3	3.2E3	482	98	2.9E3	19
A375_SM : CADD_score -	6.7E3	3.2E3	3.3E3	448	172	5.3E3	39
1000G_p3 : dbNSFP -	2E3	676	751	2E5	4.3E4	2.7E3	374
NA12878_WGS : dbNSFP -	1.3E3	628	727	1.1E4	2.7E3	1.7E3	76
NA12878_WES : dbNSFP -	1.3E3	530	516	1.8E3	242	1.6E3	21
A375_chip : dbNSFP -	1.3E3	509	546	1.8E3	441	1.7E3	9
NA12878_Amp : dbNSFP -	1.3E3	540	573	714	58	1.2E3	18
A375_SM : dbNSFP -	1.6E3	526	567	430	48	1.5E3	16
1000G_p3 : Cistrome_TF -	3.1E3	1.6E3	2.3E3	2E5	2.3E5	2.4E3	8.7E3
NA12878_WGS : Cistrome_TF -	536	218	256	1.3E4	1.4E4	982	951
NA12878_WES : Cistrome_TF -	219	95	102	1.1E3	1.4E3	827	87
A375_chip : Cistrome_TF -	230	81	104	2.1E3	2.3E3	893	124
NA12878_Amp : Cistrome_TF -	223	94	103	239	218	401	17
A375_SM : Cistrome_TF -	225	81	106	185	159	499	19
Seconds fast slow	BEDTools	BEDOPS	BCFtools	VEP	GIGGLE	vcfanno	VarNote

Supplemental Figure S7. The runtime comparisons of interval-level annotations among six methods. Six query variant datasets (A375_SM, NA12878_Amp, A375_chip, NA12878_WES, NA12878_WGS and 1000G_p3) and four genome-wide annotation databases (Cistrome_TF, dbNSFP, CADD_score and CADD_anno) were used.



Supplemental Figure S8. The runtime comparisons between VarNote and other tools for 144 benchmarks. VarNote outperforms other method in 93.75% comparisons, with only nine evaluations are slightly behindhand in seconds for small queries or databases.



Supplemental Figure S9. Random-sweep searching algorithm based on Tabix index implemented by VarNote. Each chromosome is partitioned into bins span 128kb intervals. Query intervals belonging to separate bins will be grouped together. Tabix algorithm is used to identify and merge associated bins. VarNote chromosome-sweep algorithm is used to intersect annotation records across associated bins.

			4-thread	IS				8-thread	IS	
NA12878 WGS : CADD anno-	2.1E4	1.4E5	5.1E4	1.5E4	618	2.1E4	8.1E4	3.6E4	7.8E3	304
NA12878 WES : CADD anno -	2.1E4	1.3E4	3.7E4	1.4E4	58	2.1E4	7.1E3	2.1E4	7.1E3	31
NA12878 Amp : CADD anno -	2.1E4	1.8E3	1.6E4	3.1E3	13	2.1E4	1.1E3	6.3E3	1.5E3	8
NA12878 WGS : gnomAD -	4.1E3	1.7E4	1.8E3	1.9E3	622	4.1E3	1E4	1.3E3	940	305
NA12878 WES : gnomAD -	3.8E3	1.5E3	1.3E3	1.5E3	52	3.8E3	892	770	725	29
NA12878 Amp : gnomAD -	3.9E3	238	504	358	13	3.8E3	149	425	190	7
NA12878 WGS : dbNSFP -	596	4.1E3	945	367	24	597	2.8E3	672	180	15
NA12878 WES : dbNSFP -	500	715	582	277	7	496	464	377	148	5
NA12878_Amp : dbNSFP -	500	240	532	169	6	501	165	484	93	4
Seconds			1_throad					2_throad	le.	
fast slow			1-uneau	u				2-unreau	15	
NA12878 WGS : CADD anno -	2.1E4	3.2E5	9.6E4	5.3E4	2E3	2.1E4	2.3E5	6.2E4	2.7E4	1.2E3
NA12878_WGS : CADD_anno - NA12878_WES : CADD_anno -	2.1E4 2.1E4	3.2E5 2.8E4	9.6E4 1.1E5	5.3E4 4.3E4	2E3 404	2.1E4 2.1E4	2.3E5 2E4	6.2E4 6.1E4	2.7E4 2.7E4	1.2E3 111
hast slow NA12878_WGS : CADD_anno - NA12878_WES : CADD_anno - NA12878_Amp : CADD_anno -	2.1E4 2.1E4 2.1E4	3.2E5 2.8E4 5E3	9.6E4 1.1E5 3.1E4	5.3E4 4.3E4 9.8E3	2E3 404 102	2.1E4 2.1E4 2.1E4	2.3E5 2E4 2.9E3	6.2E4 6.1E4 1.7E4	2.7E4 2.7E4 5.5E3	1.2E3 111 22
hast slow NA12878_WGS : CADD_anno - NA12878_WES : CADD_anno - NA12878_Amp : CADD_anno - NA12878_WGS : gnomAD -	2.1E4 2.1E4 2.1E4 4.2E3	3.2E5 2.8E4 5E3 4.1E4	9.6E4 1.1E5 3.1E4 4.5E3	5.3E4 4.3E4 9.8E3 6E3	2E3 404 102 1.9E3	2.1E4 2.1E4 2.1E4 4.3E3	2.3E5 2E4 2.9E3 2.8E4	6.2E4 6.1E4 1.7E4 2.4E3	2.7E4 2.7E4 5.5E3 3.2E3	1.2E3 111 22 1.1E3
hast slow NA12878_WGS : CADD anno - NA12878_WES : CADD_anno - NA12878_Amp : CADD_anno - NA12878_WGS : gnomAD - NA12878_WES : gnomAD -	2.1E4 2.1E4 2.1E4 4.2E3 3.9E3	3.2E5 2.8E4 5E3 4.1E4 3.5E3	9.6E4 1.1E5 3.1E4 4.5E3 4.4E3	5.3E4 4.3E4 9.8E3 6E3 4.7E3	2E3 404 102 1.9E3 169	2.1E4 2.1E4 2.1E4 4.3E3 4.3E3	2.3E5 2E4 2.9E3 2.8E4 2.5E3	6.2E4 6.1E4 1.7E4 2.4E3 2.3E3	2.7E4 2.7E4 5.5E3 3.2E3 2.7E3	1.2E3 111 22 1.1E3 105
Iast slow NA12878_WGS: CADD_anno - NA12878_WES: CADD_anno - NA12878_Amp: CADD_anno - NA12878_WGS: gnomAD - NA12878_WES: gnomAD - NA12878_MES: gnomAD -	2.1E4 2.1E4 2.1E4 4.2E3 3.9E3 3.8E3	3.2E5 2.8E4 5E3 4.1E4 3.5E3 633	9.6E4 1.1E5 3.1E4 4.5E3 4.4E3 1.4E3	5.3E4 4.3E4 9.8E3 6E3 4.7E3 1.2E3	2E3 404 102 1.9E3 169 36	2.1E4 2.1E4 2.1E4 4.3E3 4.3E3 3.9E3	2.3E5 2E4 2.9E3 2.8E4 2.5E3 375	6.2E4 6.1E4 1.7E4 2.4E3 2.3E3 824	2.7E4 2.7E4 5.5E3 3.2E3 2.7E3 567	1.2E3 111 22 1.1E3 105 21
NA12878_WGS : CADD_anno - NA12878_WES : CADD_anno - NA12878_WES : CADD_anno - NA12878_Mmp : CADD_anno - NA12878_WGS : gnomAD - NA12878_WES : gnomAD - NA12878_Mp : gnomAD - NA12878_WGS : dbNSFP -	2.1E4 2.1E4 2.1E4 4.2E3 3.9E3 3.8E3 602	3.2E5 2.8E4 5E3 4.1E4 3.5E3 633 8.8E3	9.6E4 1.1E5 3.1E4 4.5E3 4.4E3 1.4E3 1.7E3	5.3E4 4.3E4 9.8E3 6E3 4.7E3 1.2E3 976	2E3 404 102 1.9E3 169 36 95	2.1E4 2.1E4 2.1E4 4.3E3 4.3E3 3.9E3 602	2.3E5 2E4 2.9E3 2.8E4 2.5E3 375 6.3E3	6.2E4 6.1E4 1.7E4 2.4E3 2.3E3 824 1E3	2.7E4 2.7E4 5.5E3 3.2E3 2.7E3 567 568	1.2E3 111 22 1.1E3 105 21 46
Iast slow NA12878_WGS: CADD_anno- NA12878_WES: CADD_anno- NA12878_Amp: CADD_anno- NA12878_MGS: gnomAD- NA12878_WGS: gnomAD- NA12878_WGS: dbNSFP- NA12878_WGS: dbNSFP- NA12878_WES: dbNSFP-	2.1E4 2.1E4 2.1E4 4.2E3 3.9E3 3.8E3 602 497	3.2E5 2.8E4 5E3 4.1E4 3.5E3 633 8.8E3 1.5E3	9.6E4 1.1E5 3.1E4 4.5E3 4.4E3 1.4E3 1.7E3 1.7E3	5.3E4 4.3E4 9.8E3 6E3 4.7E3 1.2E3 976 856	2E3 404 102 1.9E3 169 36 95 20	2.1E4 2.1E4 2.1E4 4.3E3 4.3E3 3.9E3 602 495	2.3E5 2E4 2.9E3 2.8E4 2.5E3 375 6.3E3 1.1E3	6.2E4 6.1E4 1.7E4 2.4E3 2.3E3 824 1E3 960	2.7E4 2.7E4 5.5E3 3.2E3 2.7E3 567 568 474	1.2E3 111 22 1.1E3 105 21 46 11
Iast slow NA12878_WGS: CADD_anno- NA12878_WES: CADD_anno- NA12878_Amp: CADD_anno- NA12878_MGS: gnomAD- NA12878_WGS: gnomAD- NA12878_WGS: dbNSFP- NA12878_WGS: dbNSFP- NA12878_Amp: dbNSFP- NA12878_Amp: dbNSFP-	2.1E4 2.1E4 2.1E4 4.2E3 3.9E3 3.8E3 602 497 494	3.2E5 2.8E4 5E3 4.1E4 3.5E3 633 8.8E3 1.5E3 560	9.6E4 1.1E5 3.1E4 4.5E3 4.4E3 1.4E3 1.7E3 1.7E3 1.1E3	5.3E4 4.3E4 9.8E3 6E3 4.7E3 1.2E3 976 856 543	2E3 404 102 1.9E3 169 36 95 20 30	2.1E4 2.1E4 2.1E4 4.3E3 4.3E3 3.9E3 602 495 494	2.3E5 2E4 2.9E3 2.8E4 2.5E3 375 6.3E3 1.1E3 355	6.2E4 6.1E4 1.7E4 2.4E3 2.3E3 824 1E3 960 642	2.7E4 2.7E4 5.5E3 3.2E3 2.7E3 567 568 474 281	1.2E3 111 22 1.1E3 105 21 46 11 10

Supplemental Figure S10. The runtime comparisons of variant annotation using multi-threading. Three variant call results as query datasets (NA12878_Amp, NA12878_WES and NA12878_WGS) and three genome-wide annotation databases (dbNSFP, gnomAD and CADD_anno) were used. VarNote-tbi is the new multi-threading algorithm only based on Tabix index.



Supplemental Figure S11. The boxplot of rank ratio for each PICS causal variant in the GWAS signals across 16 ENCODE cell types and six different prioritization methods, the rank ratio is measured by the rank of observed variant/total number of investigated variants (including extended highly-linked variant in LD) in each GWAS signal.

Supplemental Tables

Table S1. The query datasets and annotation databases for performance evaluation	14
Table S2. The software and running parameters for interval-level overlap annotation .	15
Table S3. Summary information of 144 runtime comparisons	16
Table S4. The running parameters used in the evaluation of variant-level annotation	19
Table S5. The curation list of rare pathogenic regulatory variants	20
Table S6. The pseudocode of VarNote random-sweep searching algorithm	22

Supplemental Table S1. The query datasets and annotation databases used in the performance evaluation

	Dataset	Source
	1000G_p3	Variant call result of 1000 Genomes project phase3: <u>ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/release/20130502/ALL.wgs.phase3_shapeit2_m_vncall_integrated_v5b.20130502.sites.vcf.gz</u>
	NA12878_WGS	Variant call result of 10X Genomics Chromium whole genome sequencing for NA12878 from GIAB: <u>ftp://ftp-</u> <u>trace.ncbi.nlm.nih.gov/giab/ftp/data/NA12878/10Xgenomics_ChromiumGenome_LongR</u> <u>anger2.1_09302016/NA12878_hg19/NA12878_hg19_phased_variants.vcf.gz</u>
/ Datasets	NA12878_WES	Variant call result of Nextera Rapid Capture Exome and Expanded Exome whole exome sequencing for NA12878 from GIAB: <u>ftp://ftp-</u> <u>trace.ncbi.nlm.nih.gov/giab/ftp/data/NA12878/Garvan_NA12878_HG001_HiSeq_Exome</u> /project.NIST.hc.snps.indels.vcf
Query	NA12878_Amp	Variant call result of Ion Ampliseq Exome capture sequencing data for NA12878 from GIAB: <u>ftp://ftp-</u> <u>trace.ncbi.nlm.nih.gov/giab/ftp/data/NA12878/analysis/IonTorrent_TVC_04302015/TSV</u> <u>C_variants.vcf</u>
	A375_chip	Genotype call result of Affymetrix Genome-Wide Human SNP Array 6.0 data for A375 cell line from GDSC: <u>https://www.cancerrxgene.org/</u>
	A375_SM	Somatic mutation call result of whole exome sequencing data for A375 cell line from GDSC: https://www.cancerrxgene.org/
	CADD_score	CADD v1.3 raw score and PHRED score for all possible SNVs (80G): https://krishna.gs.washington.edu/download/CADD/v1.3/whole_genome_SNVs.tsv.gz
atabases	CADD_anno	CADD v1.3 all annotations for all possible SNVs (344G): https://krishna.gs.washington.edu/download/CADD/v1.3/whole_genome_SNVs_inclAnn o.tsv.gz
tion Da	dbNSFP	dbNSFP v3.4a all functional predictions (16G): <u>ftp://dbnsfp@dbnsfp@odbnsfp.softgenetics.com/dbNSFPv3.4a.zip</u>
Annota	gnomAD	gnomAD r2.0.2 Genomes all known variants (86G): https://gnomad.broadinstitute.org/
7	Cistrome_TF	CistromeDB 20181120 aggregated human transcription regulator ChIP-seq peaks (12G): <u>http://cistrome.org/db</u>

Tool	Version	Command line
BCFtools	1.6 (using htslib 1.6)	bcftools annotate -a {DATABASE FILE} -c CHROM,FROM,TO,ANN -h {HEADER FILE}threads 1 {QUERY FILE}
BEDOPS	2.4.37 (megarow)	bedops -i {QUERY FILE} {DATABASE FILE}
BEDTools	v2.27.1	bedtools intersect -wa -wb -a {QUERY FILE} -b {DATABASE File} -sorted
GIGGLE	v0.6.3	giggle search -i {DATABASE INDEX DIRECTORY} -q {QUERY FILE} -o -v
vcfanno	0.2.8 (built with go1.8)	vcfanno -p 1 {DATABASE CONFIGURE FILE} {QUERY FILE}
VEP	91.1	vepdir {VEP DATABASE DIRECTORY}assembly GRCh37vcfformat vcf -i {QUERY FILE} -custom {DATABASE FILE},ANN,bed,overlap,1offlinefork 1
VarNote	v1.1	java -jar VarNote.jar Intersect -Q {QUERY FILE} -D {DATABASE FILE} -T 1

Supplemental Table S2. The software information and running parameters used in the evaluation of interval-level overlap annotation

Supplemental Table S3. Summary information of 144 runtime comparisons between VarNote and other methods

Comparison	Test datasets	VarNote runtime (seconds)	Compared tool runtime (seconds)	Speedup
VarNote vs. BCFtools	1000G_p3 vs. CADD anno	11356.00	22665.00	2.00
VarNote vs. BEDOPS	1000G p3 vs. CADD anno	11356.00	21895.00	1.93
VarNote vs. BEDTools	1000G p3 vs. CADD anno	11356.00	138318.00	12.18
VarNote vs. GIGGLE	1000G_p3 vs. CADD_anno	11356.00	624065.00	54.95
VarNote vs. VEP	1000G p3 vs. CADD anno	11356.00	618069.52	54.43
VarNote vs. vcfanno	1000G p3 vs. CADD anno	11356.00	150495.00	13.25
VarNote vs. BCFtools	1000G p3 vs. CADD score	2492.52	3196.39	1.28
VarNote vs. BEDOPS	1000G p3 vs. CADD score	2492.52	3316.72	1.33
VarNote vs. BEDTools	1000 p3 vs. CADD score	2492.52	13386.00	5.37
VarNote vs. GIGGLE	1000G p3 vs. CADD score	2492.52	307813.00	123.49
VarNote vs. VEP	1000G p3 vs. CADD score	2492.52	53617.49	21.51
VarNote vs. vcfanno	1000G p3 vs. CADD score	2492.52	33408.05	13.40
VarNote vs. BCEtools	1000G n3 vs Cistrome TF	8681.00	2343 35	0.27
VarNote vs. BEDOPS	1000G p3 vs. Cistrome TF	8681.00	1613 43	0.19
VarNote vs. BEDTools	1000G p3 vs. Cistrome TF	8681.00	3138.06	0.19
VarNote vs. GIGGLE	1000G p3 vs. Cistrome_TT	8681.00	234685.00	27.03
VarNote vs. UED	1000G_p3 vs. Cistrome_TT	8681.00	202807.00	27.03
VarNote vs. vEP	1000G_p5 vs. Cistrome_IF	8681.00	202897.00	23.37
Varinole vs. vcianno	1000G_p5 vs. Cistrome_IF	8081.00	2438.10	0.28
Varinole VS. BUFIOOIS	1000G p3 VS. dbNSFP	5/4.20	/ 30.09	2.01
varNote vs. BEDOPS	1000G_p3 vs. dbNSFP	374.20	0/5.86	1.81
varinote vs. BED fools	1000G_p3 vs. dbNSFP	374.20	2007.84	5.37
VarNote vs. GIGGLE	$1000G_{p3}$ vs. dbNSFP	374.20	42901.00	114.65
VarNote vs. VEP	$1000G_{p3}$ vs. dbNSFP	374.20	202757.00	541.84
VarNote vs. vcfanno	1000G_p3 vs. dbNSFP	374.20	2661.92	7.11
VarNote vs. BCFtools	A375_SM vs. CADD_anno	75.99	22865.00	300.89
VarNote vs. BEDOPS	A375_SM vs. CADD_anno	75.99	21054.00	277.06
VarNote vs. BEDTools	A375_SM vs. CADD_anno	75.99	79781.00	1049.89
VarNote vs. GIGGLE	A375_SM vs. CADD_anno	75.99	270.60	3.56
VarNote vs. VEP	A375_SM vs. CADD_anno	75.99	4338.00	57.09
VarNote vs. vcfanno	A375_SM vs. CADD_anno	75.99	38313.00	504.18
VarNote vs. BCFtools	A375_SM vs. CADD_score	38.99	3321.12	85.18
VarNote vs. BEDOPS	A375_SM vs. CADD_score	38.99	3215.11	82.46
VarNote vs. BEDTools	A375_SM vs. CADD_score	38.99	6656.00	170.71
VarNote vs. GIGGLE	A375_SM vs. CADD_score	38.99	171.71	4.40
VarNote vs. VEP	A375 SM vs. CADD score	38.99	447.71	11.48
VarNote vs. vcfanno	A375 SM vs. CADD ^{score}	38.99	5344.13	137.06
VarNote vs. BCFtools	A375 SM vs. Cistrome TF	19.04	106.42	5.59
VarNote vs. BEDOPS	A375 SM vs. Cistrome TF	19.04	80.52	4.23
VarNote vs. BEDTools	A375 SM vs. Cistrome TF	19.04	224.79	11.81
VarNote vs. GIGGLE	A375 SM vs. Cistrome TF	19.04	158.60	8.33
VarNote vs. VEP	A375 SM vs. Cistrome TF	19.04	185.04	9.72
VarNote vs. vcfanno	A375 SM vs. Cistrome TF	19.04	499.15	26.22
VarNote vs. BCFtools	A375 SM vs. dbNSFP	15.65	567.43	36.26
VarNote vs. BEDOPS	A375 SM vs. dbNSFP	15.65	526.01	33.61
VarNote vs. BEDTools	A375 SM vs. dbNSFP	15.65	1570.74	100.37
VarNote vs. GIGGI F	A375 SM vs. dbNSFP	15.65	47.72	3 05
VarNote vs. VFP	A375 SM vs. dbNSFP	15.65	429.66	27 45
VarNote vs. vcfanno	A375 SM vs. dbNSFP	15.65	1528.87	97.60
VarNote vs. BCFtools	A375 chin ve CADD anno	500 43	21648.00	36.11
VarNote vs. BEDODS	A375 chin vs. CADD_anno	500 /2	20461.00	2/ 12
VarNote vs. BEDOCS	A375 chin vs. CADD_anno	500 /2	74452.00	10/ 20
VarNote vs. CICCLE	A375 chip vs. CADD_allilo	500 42	2500 85	124.20
VarNote vs. GIGGLE	AS / 5_ cmp vs. CADD_anno	500 42	2300.83	4.1/
Vallinute vs. VEP	AS / 3_ cnip vs. CADD_anno	599.43	5/515.00	95.61
Varinote vs. vcfanno	A3/5_cnip vs. CADD_anno	599.43	1006/2.00	167.95
varinote vs. BCFtools	A3/5_chip vs. CADD_score	146.93	3207.64	21.83
varNote vs. BEDOPS	A3/5_chip vs. CADD_score	146.93	3020.57	20.56
VarNote vs. BEDTools	A375_chip vs. CADD_score	146.93	6161.00	41.93
VarNote vs. GIGGLE	A375_chip vs. CADD_score	146.93	1570.83	10.69
VarNote vs. VEP	A375_chip vs. CADD_score	146.93	5398.00	36.74
VarNote vs. vcfanno	A375_chip vs. CADD_score	146.93	12372.27	84.21
VarNote vs. BCFtools	A375_chip vs. Cistrome_TF	124.04	103.91	0.84
VarNote vs. BEDOPS	A375_chip vs. Cistrome_TF	124.04	80.70	0.65
V M OFDT 1	A 275 shin va Cistuana TE	124.04	230.20	1.86

VarNote vs. GIGGLE	A375_chip vs. Cistrome_TF	124.04	2345.14	18.9
varNote vs. VEP	A3/5_chip vs. Cistrome_TF	124.04	2123.99	17.1
VarNote vs. vcfanno	A3/5_chip vs. Cistrome_TF	124.04	892.76	7.2
VarNote vs. BCFtools	A3/5_chip vs. dbNSFP	8.71	546.31	62.7
VarNote vs. BEDOPS	A375_chip vs. dbNSFP	8.71	509.24	58.4
VarNote vs. BEDTools	A375_chip vs. dbNSFP	8.71	1268.80	145.6
VarNote vs. GIGGLE	A375_chip vs. dbNSFP	8.71	441.38	50.6
VarNote vs. VEP	A375_chip vs. dbNSFP	8.71	1799.69	206.6
VarNote vs. vcfanno	A375_chip vs. dbNSFP	8.71	1715.83	197.0
VarNote vs. BCFtools	NA12878_Amp vs. CADD_anno	49.66	22689.00	456.8
VarNote vs. BEDOPS	NA12878_Amp vs. CADD_anno	49.66	21680.00	436.5
VarNote vs. BEDTools	NA12878_Amp vs. CADD_anno	49.66	75198.00	1514.2
VarNote vs. GIGGLE	NA12878_Amp vs. CADD_anno	49.66	151.46	3.0
VarNote vs. VEP	NA12878 Amp vs. CADD anno	49.66	5637.00	113.5
VarNote vs. vcfanno	NA12878 Amp vs. CADD anno	49.66	27697.00	557.7
VarNote vs. BCFtools	NA12878 Amp vs. CADD score	19.18	3232.38	168.5
VarNote vs. BEDOPS	NA12878 Amp vs. CADD score	19.18	3200.88	166.8
VarNote vs. BEDTools	NA12878 Amp vs. CADD score	19.18	6362.00	331.7
VarNote vs. GIGGLE	NA12878 Amp vs CADD score	19.18	97.80	51
VarNote vs. VEP	NA12878 Amp vs. CADD_score	19.18	481 52	25.1
VarNote vs. vcfanno	NA12878 Amp vs. CADD score	19.18	2871 12	149.6
VarNote vs. PCEtools	NA12878 Amp vs. Cistrome TE	17.10	102.88	5.0
VarNote vs. DEPt0015	NA12878 Amp vs. Cistrome_IT	17.42	04.41	5.9
VarNote vs. BEDOFS	NA12878 Amp vs. Cistrome_IT	17.42	222.01	12.9
VarNote vs. BEDTOOIS	NA12878 Amp vs. Cistrome_TF	17.42	225.01	12.0
Varinole vs. GIGGLE	NA12878_Amp vs. Cistrome_IF	17.42	210.17	12.3
VarNote vs. VEP	NA128/8_Amp vs. Cistrome_IF	17.42	238.72	13./
VariNote vs. vctanno	NA128/8_Amp vs. Cistrome_IF	17.42	400.80	23.0
VarNote vs. BCFtools	NA12878_Amp vs. dbNSFP	17.62	5/3.18	32.5
VarNote vs. BEDOPS	NA12878_Amp vs. dbNSFP	17.62	539.58	30.6
VarNote vs. BEDTools	NA12878_Amp vs. dbNSFP	17.62	1340.92	76.1
VarNote vs. GIGGLE	NA12878_Amp vs. dbNSFP	17.62	58.25	3.3
VarNote vs. VEP	NA12878_Amp vs. dbNSFP	17.62	714.00	40.5
VarNote vs. vcfanno	NA12878_Amp vs. dbNSFP	17.62	1192.21	67.6
VarNote vs. BCFtools	NA12878_WES vs. CADD_anno	352.47	22544.00	63.9
VarNote vs. BEDOPS	NA12878_WES vs. CADD_anno	352.47	21747.00	61.7
VarNote vs. BEDTools	NA12878_WES vs. CADD_anno	352.47	74965.00	212.6
VarNote vs. GIGGLE	NA12878 WES vs. CADD anno	352.47	1153.61	3.2
VarNote vs. VEP	NA12878 WES vs. CADD anno	352.47	29828.00	84.6
VarNote vs. vcfanno	NA12878 WES vs. CADD anno	352.47	102118.00	289.7
VarNote vs. BCFtools	NA12878 WES vs. CADD score	142.37	2917.42	20.4
VarNote vs. BEDOPS	NA12878 WES vs. CADD score	142.37	3212.17	22.5
VarNote vs. BEDTools	NA12878 WES vs. CADD score	142.37	6048.00	42.4
VarNote vs. GIGGLE	NA12878 WES vs. CADD score	142.37	694.71	4.8
VarNote vs. VEP	NA12878 WES vs. CADD score	142.37	2400 67	16.8
VarNote vs. vcfanno	NA12878 WES vs. CADD_score	142.37	10201 37	71.6
VarNote vs. RCEtools	NA12878 WES vs. CADD_SOLE	87.26	102 27	/1.0
VarNote ve REDODC	NA12070_WES VS. CISUOME_IF	87.20	05.00	1.1
VarNote va. DEDUPS	NA12070_WES vs. Cistrome_IF	87.20	23.07	1.0
VarNote vs. BED1001S	NA120/0_WES VS. UISTOME_IF	01.20	∠10.88 1292-11	2.3
VarNote vs. GIGGLE	NA12070_WES VS. UIStrome_IF	0/.20	1383.11	15.8
VarNote vs. VEP	NA128/8_WES VS. UIStrome_IF	87.20	1000.25	12.2
varivote vs. vctanno	NA128/8_WES vs. Cistrome_TF	87.26	827.16	9.4
VarNote vs. BCFtools	NA12878_WES vs. dbNSFP	21.38	515.86	24.1
varNote vs. BEDOPS	NA128/8_WES vs. dbNSFP	21.38	530.14	24.8
VarNote vs. BEDTools	NA12878_WES vs. dbNSFP	21.38	1255.68	58.7
VarNote vs. GIGGLE	NA12878_WES vs. dbNSFP	21.38	242.23	11.3
VarNote vs. VEP	NA12878_WES vs. dbNSFP	21.38	1807.31	84.5
VarNote vs. vcfanno	NA12878_WES vs. dbNSFP	21.38	1562.93	73.1
VarNote vs. BCFtools	NA12878_WGS vs. CADD_anno	1960.29	25960.75	13.2
VarNote vs. BEDOPS	NA12878_WGS vs. CADD_anno	1960.29	21744.00	11.0
VarNote vs. BEDTools	NA12878_WGS vs. CADD_anno	1960.29	75880.00	38.7
VarNote vs. GIGGLE	NA12878 WGS vs. CADD anno	1960.29	13855.00	7.0
VarNote vs. VEP	NA12878 WGS vs. CADD anno	1960.29	331010.00	168.8
VarNote vs. vcfanno	NA12878 WGS vs. CADD anno	1960.29	98069.00	50.0
VarNote vs. BCFtools	NA12878 WGS vs. CADD score	1004.61	3510.78	3 4
VarNote vs BEDOPS	NA12878 WGS vs CADD score	1004.61	3257.67	3.1
VarNote vs. BEDOI 5	NA12878 WGS vs. CADD score	1004.61	6191.00	5.2
VarNote vs. GIGCI E	NA12878 WCS VS. CADD SCORE	1004.01	7050.00	7.0
VarNote vs. VED	NA12878 WGS VS. CADD_SCORE	1004.01	28002 00	7.9 ד סר
Vallaule vs. VEP	NA12070 WG5 VS. CADD_score	1004.01	26902.00	28.7
varinote vs. vcianno	NA128/8_WGS VS. CADD_score	1004.61	13301.17	15.2

VarNote vs. BCFtools	NA12878 WGS vs. Cistrome TF	951.34	255.86	0.27
VarNote vs. BEDOPS	NA12878 WGS vs. Cistrome TF	951.34	218.20	0.23
VarNote vs. BEDTools	NA12878 WGS vs. Cistrome TF	951.34	535.84	0.56
VarNote vs. GIGGLE	NA12878 WGS vs. Cistrome TF	951.34	14432.00	15.17
VarNote vs. VEP	NA12878 WGS vs. Cistrome TF	951.34	13163.00	13.84
VarNote vs. vcfanno	NA12878 WGS vs. Cistrome TF	951.34	981.81	1.03
VarNote vs. BCFtools	NA12878_WGS vs. dbNSFP	76.03	727.30	9.57
VarNote vs. BEDOPS	NA12878 WGS vs. dbNSFP	76.03	627.57	8.25
VarNote vs. BEDTools	NA12878_WGS vs. dbNSFP	76.03	1310.52	17.24
VarNote vs. GIGGLE	NA12878 WGS vs. dbNSFP	76.03	2652.47	34.89
VarNote vs. VEP	NA12878_WGS vs. dbNSFP	76.03	11439.00	150.45
VarNote vs. vcfanno	NA12878 WGS vs. dbNSFP	76.03	1704.22	22.42

Tool	Database dataset	Database format	Extracted Features	Command line
BCEtools	dbNSFP	BED	SIFT_score, Polyphen2_HDIV_score, MutationAssessor_score, M-CAP_score, REVEL score	bcftools annotate -a {DATABASE FILE} -c {BED FIELDS} -h {HEADER LINES}threads {THREADS} {QUERY FILE}
Der tools	gnomAD	VCF	AF, AN, GC, CSQ	bcftools annotate -a {DATABASE FILE} -c {FIELDS}threads {THREADS} {QUERY FILE}
	CADD_anno	BED	GerpRS, fitCons, RawScore, PHRED	bcftools annotate -a {DATABASE FILE} -c {BED FIELDS} -h {HEADER LINES}threads {THREADS} {QUERY FILE}
dbNSFP VarNote		BED	SIFT_score, Polyphen2_HDIV_score, MutationAssessor_score, M-CAP_score, REVEL_score	java -jar VarNote.jar Annotation -Q {QUERY FILE} - D:db,mode=1 {DATABASE FILE} -T {THREADS} -
vcfanno	gnomAD	VCF	AF, AN, GC, CSQ	A {FIELDS}
	CADD_anno	BED	GerpRS, fitCons, RawScore, PHRED	
	dbNSFP BED		SIFT_score, Polyphen2_HDIV_score, MutationAssessor_score, M-CAP_score, REVEL_score	vcfanno -p {THREADS} {CONFIGURE FILE} {QUERY FILE}
	gnomAD	VCF	AF, AN, GC, CSQ	
	CADD_anno	BED	GerpRS, fitCons, RawScore, PHRED	-
	dbNSFP	BED	SIFT_score, Polyphen2_HDIV_score, MutationAssessor_score, M-CAP_score, REVEL_score	vepdir {VEP DATABASE}assembly GRCh37 vcfformat vcf -i {QUERY FILE} -custom {DATABASE FILE},{DATABASE ID},bed,exact,0 offlinefork {THREADS}
VEP	gnomAD	VCF	AF, AN, GC	vepdir {VEP DATABASE}assembly GRCh37 vcfformat vcf -i {QUERY FILE} -custom {DATABASE FILE},{DATABASE ID},vcf,exact,0,{DATABASE FIELDS}offline fork {THREADS}
	CADD_anno	BED	GerpRS, fitCons, RawScore, PHRED	vepdir {VEP DATABASE}assembly GRCh37 vcfformat vcf -i {QUERY FILE} -custom {DATABASE FILE},{DATABASE ID},bed,exact,0 offlinefork {THREADS}

Supplemental Table S4. The running parameters used in the evaluation of variant-level annotation

CHR	POS	rsID	REF	ALT	Disease	Functional Assays	Molecular Effect	Source	PMID
1	11905784	rs12565	Т	G	Familial atrial fibrillation	EMSA, luciferase reporter assay (HEK293)	decreases binding affinity for REST to prompt NPPA expression	HGMD	22496669
1	101184877	rs3783605	А	G	Thromboembolic diseases, asthma, and multiple myeloma	ChIP (EBV-transformed lymphoblast), luciferase reporter assay (Jurkat)	increases binding affinity for ETS2	HGMD	17431880
1	209878286	rs13306421	G	А	Metabolic syndrome	luciferase reporter assay (CHO)	prompts HSD11B1 expression	HGMD	19934376
3	9791918	rs56387615	G	С	Gastric cancer	luciferase reporter assay (HEK293, HeLa)	decreases promoter activity to repress OGG1 expression	HGMD	21822670
3	9791948	rs1801129	А	G	Type 2 diabetes mellitus (T2DM)	luciferase reporter assay (HeLa, HEK293)	decreases promoter activity to repress OGG1 expression	HGMD	20562008
3	9791953	rs1801126	G	Т	Type II epithelial ovarian cancer (EOC)	luciferase reporter assay (HEK293)	represses OGG1 expression	HGMD	21997177
3	37035012	rs587779001	С	А	Lynch syndrome	luciferase reporter assay (HEK293)	represses MLH1 expression	ClinVar, Genomiser	21840485
5	147211355	rs191068215	С	Т	Chronic pancreatitis	luciferase reporter assay (AR42J, 266-6, HEK293T)	decreases promoter activity to repress SPINK1 expression	NCBoost, HGMD	25792561
6	1610252	rs77888940	С	G	Primary congenital glaucoma	luciferase reporter assay (HEK293T)	increases protein levels and transactivation activity of FOXC1 by disrupt a consensus sequence for a terminal oligopyrimidine tract	ClinVar, Genomiser	26220699
6	88299677	rs201150141	Т	С	Pontocerebellar hypoplasia	luciferase reporter assay (HEK293T)	decreases promoter activity to repress RARS2 expression	ClinVar, Genomiser, CDTS	25809939
11	5248393	rs34883338	G	А	Familial hypercholesterolemia	luciferase reporter assay (HepG2), EMSA (SL-2)	decreases binding affinity for Sp1 and decreases promoter activity to repress LDLR	NCBoost, HGMD, CDTS	11792717
11	102596480	rs11225395	А	С	Preterm premature rupture of the membranes	luciferase reporter assay (BeWo, HTR-8/Svneo, JEG-3), EMSA (BeWo)	decreases binding affinity for unknown TF(s) and increases promoter activity to prompts MMP8 expression	HGMD	15367487
12	52308396	rs201378973	G	С	Hereditary hemorrhagic telangiectasia (HHT)	luciferase reporter assay (HUVEC), EMSA (HUVEC)	decreases binding affinity for Sp1 to repress ACVRL1 expression	NCBoost	23460919
16	67520735	rs8047574	С	Т	Low body fatness	luciferase reporter assay (NCI-h295R)	increases promoter activity to prompt AGRP expression	HGMD	15121772
16	72088461	rs5471	А	С	Ahaptoglobinaemia	luciferase reporter assay (HepG2)	decreases promoter activity to repress HP expression	HGMD	14616769

Supplemental Table S5. The curation list of rare pathogenic regulatory variants and experimental validation evidence

17	69108655	rs1859961	А	G	Prostate cancer	ChIP (LNCaP), luciferase reporter assayq (LNCaP)	decreases binding affinity for FOXA1, increases binding affinity for AP-1 and increases enhancer activity to prompt SOX9 expression	HGMD	22665440
19	49468616	rs398124638	G	С	Hereditary hyperferritinaemia- cataract syndrome	EMSA (K562)	decreases binding affinity for iron-regulatory proteins	ClinVar, Genomiser, CDTS	10759702
20	23030292	rs16984852	С	А	Venous thrombosis	luciferase reporter assay (HUVECs, HEK293, COS-7)	repress THBD expression	ClinVar, Genomiser	23332921

Supplemental Table S6. The pseudocode of VarNote random-sweep searching algorithm

<u>Main:</u> read input file map input queries into groups according to given CPU threads foreach group do **SearchInGroup** end reduce group results to final output

<u>SearchInGroup:</u> foreach query in group do foreach database do **SearchHitsInDatabase** end merge hits for each query end

SearchHitsInDatabase:

if query is first read in chromosome then foreach database do **loadIndexForChr** end

foreach SROB in index do if query.end < SROB.min then nextBlock elseif query.beg <= SROB.max then do SearchHitsInBlock break loop end

SearchHitsInBlock:

foreach linklist.record in linklist do

if linklist.record.end < query.beg then remove linklist.record from linklist elseif linklist.record.beg < query.end then **RandomAccessAnno** from linklist.record end

while (next ROB in Block from pointer is not null)

do

if ROB.record.end <= query.beg then continue

elseif ROB.record.beg < query.end then **RandomAccessAnno** and put ROB.record into linklist

else set a pointer to current ROB end

* loadIndexForChr: will load all SROBs for current chromosome from index file

* RandomAccessAnno: will randomly access annotations from OB of database file

Supplemental Codes

Supplemental_Code.zip: The file contains all scripts and source codes for methods evaluation, manuscript results and software functions.

All latest source code and scripts for methods evaluation and manuscript results are also available at <u>https://github.com/mulinlab/VarNote</u>