STICI: Split-Transformer with Integrated Convolutions for Imputation

Mohammad Erfan Mowlaei[1], Chong Li[1], Oveis Jamialahmadi[2], Raquel Dias[3], Junjie Chen[4], Benyamin Jamialahmadi[5], Timothy Richard Rebbeck[6,7], Vincenzo Carnevale[8,9], Sudhir Kumar[1,8,10], Xinghua Shi[1,8*]

[1*]Computer & Information Sciences, College of Science and Technology, Temple University, 1925 N. 12th Street, Philadelphia, 19122, PA, USA.
[2]Department of Molecular and Clinical Medicine, Institute of Medicine, Sahlgrenska Academy, Wallenberg Laboratory, University of Gothenburg, Gothenburg, Sweden.
[3]Department of Microbiology and Cell Science, University of Florida, 1355 Museum Dr, Gainesville, 32603, FL, USA.
[4]Computer Science and Technology, Harbin Institute of Technology, Shenzhen University Town, Shenzhen, 518055, Guangdong, China.
[5]David R. Cheriton School of Computer Science, University of Waterloo, 200 University Avenue West, Waterloo, N2L 3G1, ON, CA.
[6]Division of Population Sciences, Dana-Farber Cancer Institute, 450 Brookline Ave, Boston, 02215, MA, USA.
[7]Department of Epidemiology, Harvard T. H. Chan School of Public Health, 677 Huntington Ave, Boston, 02115, MA, USA.
[8]Institute for Genomics and Evolutionary Medicine, Temple University, 1925 N. 12th Street, Philadelphia, 19122, PA, USA.
[9]Institute for Computational Molecular Science, Temple University, 1925 N. 12th Street, Philadelphia, 19122, PA, USA.
[10]Department of Biology, Temple University, 1925 N. 12th Street, Philadelphia, 19122, PA, USA.

*Corresponding author(s). E-mail(s): mindyshi@temple.edu;
Contributing authors: mohammad.erfan.mowlaei@temple.edu; chong.li0001@temple.edu; oveis.jamialahmadi@wlab.gu.se; raquel.dias@ufl.edu; junjiechen@hit.edu.cn; B2jamial@uwaterloo.ca; timothy_rebbeck@dfci.harvard.edu; vincenzo.carnevale@temple.edu; s.kumar@temple.edu;

Abstract

Motivation: Despite recent advances in sequencing technologies, genome-scale datasets continue to have missing bases and genomic segments. Incomplete sequence data can undermine downstream analyses, such as disease risk prediction and association studies. Consequently, the imputation of missing information is a common preprocessing step for which many methodologies have been developed. However, the imputation of genotypes of certain genomic regions and variants, including large structural variants, remains a challenging problem.

Results: Here, we present a transformer-based convolutional deep learning framework, called split-transformer with integrated convolutions for imputation (STICI), for accurate genome-scale genotype

imputation. Empowered by the attention-based transformer model, STICI can be trained for any collection of genomes automatically using self-supervision. STICI handles multi-allelic genotypes naturally, unlike other models that need special treatments. STICI models automatically learned genome-wide patterns of linkage disequilibrium (LD), evidenced by much higher imputation accuracy in high LD regions. Also, STICI models trained through sporadic masking for self-supervision performed well in imputing systematically missing information. Our imputation results on the human 1000 Genomes Project and non-human genomes show that STICI can achieve high imputation accuracy comparable to the state-of-the-art genotype imputation methods, with the additional capability to impute multi-allelic variants and various types of genetic variants. Moreover, STICI showed excellent performance without needing any special presuppositions about the patterns in the underlying data when applied to collections of non-human genomes, pointing to adaptability and applications of STICI to impute missing genotypes in any species.

Keywords: Genotype, Structural variation, Imputation, Convolutional neural network, Transformer

1 Introduction

Genetic and genomic studies, such as linkage analysis, genome-wide association study (GWAS), and polygenic risk score (PRS) estimation, enable us to dissect the genetic architecture of complex traits and diseases [1]. In recent years, whole genome sequencing (WGS) platforms and genotyping techniques have become increasingly cost-effective, resulting in the accumulation of large collections of genotypes and deeper insights into the genetic architecture of individual traits and diseases.

The resolution of genotyping has been improving due to advances in genotyping and sequencing technologies, accompanied by algorithmic improvement in analyzing such data. However, genotype data still contains missing values and untyped loci [2]. Such missing genotypes may decrease the statistical power in disease association studies and causal variant discovery [3–5]. Causes of missing genotypes include the difficulty in sequencing rare alleles [6–8], failure of experimental assays, genotype calling errors, and differences in densities and properties of genotyping platforms [3]. As such, genotype missingness, as depicted in Figure 1, can be classified into two distinct categories: sporadic missingness, where for each site/segment, some values could be absent; and systematic missingness, in which some genomic loci or segments are not genotyped. Challenges in handling missing data are further compounded when considering different types of genetic variation in addition to Single Nucleotide Variants (SNVs). Compared with SNVs, Structural Variations (SVs) pose greater challenge in genotype calling and imputation due to their increased complexity, limitations of current sequencing technologies, extensive allelic diversity, and their variable frequencies among populations [9, 10]. Moreover, SVs can have a more significant impact on genetic diseases than SNVs, so their accurate imputation can lead to enhancements in disease association studies [11].

Consequently, there is a common need for reliable imputation of genotypes using computational methods. Imputation is the process of inferring missing values in the data based on the information already present in the dataset, such as the density and distribution of variants of various complexities within and among sequences in the dataset. Imputation of missing data in genomics needs specialized methods because genomic information is inherently different from data in many other domains, such as vision or natural language processing. For example, genotype data have some unique characteristics including high data dimensionality when the number of variants is usually larger than sample size, linear and nonlinear correlations among the variants [12], and shared segments of sequences due to common ancestries. Furthermore, given the multifaceted nature of genotype imputation, it is well-recognized that no single method can serve as a universal solution. Consequently, it is a common practice in the field to utilize multiple imputation tools for a particular study.

Widely-used imputation methods often require a reference panel of genomes to impute missing genotypes in other genomes, which assumes that the missing information comes from the same ancestry patterns as those in the reference panel [13]. These methods utilize Hidden Markov Models (HMMs), graphical models, and haplotype-cluster algorithms to impute missing values [14]. For example, Minimac4 [15], the most recent version of MACH [16], uses an HMM. For each individual, Minimac4 updates the phase iteratively in both directions based on haplotypes in the reference panel and neighboring loci in

Fig. 1 Two categories of genotypes missingness . a. Sporadic missingness: It arises due to genotype calling errors and assay failures. Prediction of sporadic missingness is typically done during the pre-phasing step of imputation pipelines. b. Systematic missingness: Di erences in sequencing resolution are common causes of systematic missingness because a subset of genomic positions are assayed. The inference of missing variants in untyped regions is a major focus of imputation pipelines.

the individual. It splits sequences into overlapping chunks in order to reduce memory consumption and make the model scalable. Similarly, SHAPEIT5 [17], Impute5 [18] and Beagle [19] also employ HMMs to perform imputation. The Haplotype-clustering algorithm is utilized in the fastPHASE [20] in order to cluster haplotypes in an SNV-wise manner and impute missing values per locus. GLIMPSE2 [21] uses a HMM to genotype low-coverage whole-genome sequencing (WGS) data.

Deep Learning (DL) methods have lately been applied for genotype imputation. Sparse Convolutional Denoising autoencoder (SCDA) is used in [14] to impute missing data in the Human Leukocyte Antigen (HLA) region on chromosome 6 and yeast [22] genotypes. In [3] an improvement in SCDA training is proposed to improve the model's performance. Similarly, [23] used autoencoders on identi ed linkage disequilibrium (LD) blocks as well as focal loss to improve the performance. RNN-IMP [24] utilizes recurrent neural networks (RNNs) and augments the samples using recombination and mutation in order to impute systematic missingness in genotype data. In a follow up study [25], this model was updated with a complementary denoising autoencoder to perform pre-phasing in addition to systematic imputation. GRUD [26] utilizes RNN-IMP in an adversarial training schema which substantially reduces the model training time. DEEP*HLA [27] uses a convolutional neural network to perform imputation on pre-phased genotypes at the gene level. Inspired by DEEP*HLA, HLARIMNT [28] uses transformers for the same task.

While the overall performance of existing imputation methods for genomic data is generally good for common variants and alleles with high frequency alleles, it is still challenging to impute rare variants, especially in scenarios where reference panels are limited. Also, many current solutions cannot directly handle multi-allelic variants. Moreover, existing methods have not been systematically benchmarked for imputing SVs, particularly complex variants with multiple alleles, to the best of our knowledge. Also, the training of DL models [3, 14, 23, 27] is generally slow, and they usually need many more samples to perform as well as classical methods based on HMMs, such as Minimac4 [15] and Beagle5.4 [19]. Although RNN-IMP and GRUD [24, 26] addressed this performance disparity, these methods require retraining when the variant sets in the target are di erent from those in the training set. Although GRUD has the bene t of faster model training, it is not designed to handle sporadic missingness (Figure 1.a). Finally, the rest of the existing DL methods rely on convolutional neural networks (CNNs), which excel at exploiting local patterns but do not exhibit a robust mechanism to e ectively capture pairwise correlations among local and distant markers simultaneously, such as the presence of LD blocks in genotypes.

Compared with CNNs, the emerging attention mechanism in Transformers o ers an e ective solution to capturing local and distal interactions in genomes [29]. The attention mechanism in DL mimics visual attention to focus on speci c parts of pictures [30, 31] by calculating importance scores among genomic loci. Therefore, attention can capture global interactions amongst markers. Transformers utilize multi-head attention to capture intricate and multi-level interactions among the variants. AlphaFold2 [32] and ESM-2 [33] are successful examples of transformers in biological sequence analysis.

In this article, we present a novel genotype imputation model, STICI, based on the attention mechanisms in a transformer framework and convolutional blocks. Our model utilizes attention to capture patterns among SNVs and SVs in the genome collections analyzed. We found STICI to achieve high imputation accuracy at a modest memory consumption cost, achieved by dividing the data into chunks (following [15]) that enables e cient application of STICI to long sequences. Furthermore, STICI needs to be trained only once, unlike other DL models. After this training, the imputation time in STICI is comparable or faster than using classical methods (Supplementary Tables 10 and 11). In brief, our study makes the following key contributions.

- We propose a DL transformer framework termed STICI, designed to speci cally address the genotype imputation problem.
- STICI imputation does not need a standard reference panel, which makes it more generally applicable to any data set.
- STICI excels at SV imputation, where the variants harbor a higher degree of complexity while achieving performance comparable to competing models for SNV imputation.
- We analyze the e ect of di erent masking rates on building better imputation models and explain the reasons for STICI's performance.

2 Results

2.1 Overview of the study

In this section, rst, we present the results of our empirical study to nd the optimal masking percentage (masking rate, MaskR) for STICI training in order to investigate the need for building imputation models for speci c missing rate (MissR) in the target dataset . After that, we present results for sporadic missingness imputation on the yeast dataset, SVs in human chromosome 22, and extensive SVs dataset (see Subsection 4.1 in Methods for the dataset details). This is followed by systematic missingness imputation for selected regions in human chromosome 22, simulated human chromosome 19, rat, and chicken datasets. We benchmarked STICI's performance against classical pre-phasing methods (SHAPEIT5 and Eagle2 [34]), classical imputation methods (Impute5, Beagle4, Beagle5.4, Minimac3, and Minimac4.1.4), deep learning models (SCDA and AE), and two variants of STICI, one that uses no embedding (STICI-NE) and another that uses an additional R^2 loss term (STICI-R^2). The details of STICI architecture and aforementioned methods are discussed in Subsections 4.2 and 4.3, respectively.

2.2 Optimal masking rate analysis

For this analysis, we used the HLA region on chromosome 6 from the human 1000 Genomes Project and performed a 3-fold cross-validation on the data. The aim was to examine the relationship of MaskR for the training set with varying MissR in target sets. The results are presented in Figure 2 in which the results on the left/right column belong to the validation and test set, respectively.

Figures 2 a & b show that the performances of STICI models trained using MaskR of 0.5 and 0.7 were high for imputations in which MissRs were up to 0.5. Therefore, a single STICI model, trained with a MaskR of 0.5, could be used for a variety of research datasets as long as the MissR is less than 0.5. However, STICI models trained with MaskR $>$ 0:5 are needed for reliable imputations when the target datasets have more than 50 percent missing variants. Therefore, we recommend STICI models trained with a MaskR of 0.5 for imputing sporadically missing variants and a higher MaskR (preferably matching the MissR in target dataset indicated by our empirical studies data) for other datasets.

Figures 2 a & b also show that when the target MissR is su ciently low, the performance gap of the imputation models is not discernible. The performance gap becomes evident with a MissR of 0.2 or higher. The underlying cause of this observation is that when the MissR is extremely low, a su cient number of variants in LD with the target variant are readily available, making predictions less challenging for all the models. Conversely, a large MissRs means that the amount of information from LD blocks diminishes, presenting a greater challenge to the imputation model.

Figures 2 c & d show that, generally, STICI models trained with lower MaskR will produce poor performance for imputing missing SNVs located in regions with high LD. For instance, variants in regions with LD = 0.01 have the lowest accuracy for all the masking percentages. Additionally, these results

indicate that it is easier to predict missing data in high LD regions compared to low LD regions, which aligns well with biological expectations that low LD regions do not bene t from additional information (LD) available for better imputation of high LD regions. These trends suggest that the use of a low MaskR prevents the model from learning LD patterns, resulting in a worse performance. In other words, the model training needs to e ectively disturb the LD blocks (and other latent patterns among variants) to capture direct and indirect correlations and haplotypes. Consequently, MaskR of 0.5 and higher provides robust results across a large range of target MissR values.

Fig. 2 Average accuracy over 3-fold cross-validation for validation and test sets in the HLA dataset using di erent masking rate (MaskR) values during STICI training. a. and b. A breakdown of average accuracy for various missing rate (MissR) values of validation/test set when the model is trained using di erent MaskR values. The patterns show that a model trained using a higher MaskR is more robust across di erent target MissRs. c. and d. Average accuracy for validation/test sets over 3 folds and di erent MissR values calculated for various LD bins. The trend suggests that a higher MaskR increases the performance across LD bins, which could be attributed to the impact of MaskR on STICI to learn LD patterns comprehensively. When MaskR is low, STICI imputations do not bene t from the LD patterns present and, thus, STICI does not learn the majority of pairwise correlations (LD) among the variants. Consequently, STICI is not able to infer the missing value using all possible information in the respective LD block of the target variant.

2.3 The relative performance of STICI for sporadic missingness

For each dataset in this experiment, we performed a 3-fold cross-validation where missing values were introduced using fixed random seeds to ensure reproducibility of results across experiments and methods. We also verified that there is no data leakage among the training and test sets across the folds. The missing values were distributed randomly according to one of three strategies: uniformly, based on Minor Allele Frequency (MAF), or based on LD. These methods were chosen to ensure that missing values are representative of the data distribution in different biological aspects. Further details on these procedures can be found in the methods section. In all of the experiments, missing positions in the test sets were the same for all the methods.

The overall results for the yeast and chromosome 22 datasets are presented in Supplementary Table 7. The numerical values in this table indicate the average of the metric values on the test sets in a 3-fold cross-validation. We used maximum LD bins and/or MAF bins (Figure 5 b, and c) to distribute missing positions in the datasets extracted from the human 1000 Genomes Project. If the bins had too few positions (e.g., at a 0.01 MissR on chromosome 22 datasets), we excluded this MissR for the experiments related to these datasets. We used a consistent approach to introduce missing values in chromosomes 6, 10, 16, and 20 based on LD distributions and a single test MissR of 0.2. In this experiment, we focused on comparing Minimac4.1.4, Beagle5.4, SHAPEIT5, and STICI, because they were identified as the top performers from classical and DL methods in prior experiments. We also added Eagle2 [34] to the competing methods in this experiment. We employed 3-fold cross-validation for both methods, training and imputing each chromosome separately. R^2 was calculated for each variant, and the results were averaged over each fold, chromosome, and SV type. Figure 3 presents the experimental results for the extensive structural variation datasets where the top plot shows the improvement that STICI provides compared to the best of other methods for each SV type, and is calculated as follows:

$$\text{Improvement}\,(\%) = \frac{R^2_{\text{STICI}} - R^2_{\text{Best}}}{R^2_{\text{Best}}} \quad 100$$

Yeast dataset: Missing positions in samples were selected randomly, as the LD analysis showed that the maximum LD for all the SNVs was high in the $[0.8; 1.0]$ range. As mentioned, Minimac4.1.4, SHAPEIT5, and Beagle5.4 cannot be used to impute variants of the yeast dataset due to the lack of a reference panel. However, STICI could be applied and outperformed other methods, achieving a minimum average imputation accuracy of 99.86%. Overall, all the applicable models performed well on the yeast dataset, which we attribute to the presence of high LD among SNVs in this dataset.

Deletions in chromosome 22: For this dataset, we introduced missing positions proportional to the maximum-LD/MAF distribution Figure 5.b. Overall, STICI emerged as the best or the second-best model for imputation across all the metrics. STICI was more accurate than others for LD/MAF missingness distribution schema. Furthermore, SCDA+ demonstrates a substantial performance advantage over AE in terms of IQS and R^2 in the majority of the cases. Supplementary Table 8 shows the accuracy trends for different maximum LD values for this dataset when missing values are distributed proportional to variant density in maximum LD bins. Minimac4.1.4 and Beagle5.4 were less accurate for SNVs with lower maximum LD compared to AE, SCDA+, STICI-NE, and STICI. Since HMMs and graphical models rely on conditional probabilities, we suggest that they would perform relatively weak due to a low correlation between the events (states).

All SVs in chromosome 22: Similar to the previous dataset, missing positions were distributed among SVs based on maximum-LD/MAF (Figure 5.c). Despite having a reference panel, Minimac4.1.4 and SHAPEIT5 cannot be directly used for some missing variants for this dataset because they can only handle bi-allelic events. Furthermore, IQS is not well-defined for multi-allelic events.

Supplementary Table 7 shows that STICI outperforms all other methods on average accuracy and F1-score. STICI performance in terms of R^2 is much better than the competing methods at high MissRs. R^2 considers the correlation among genotypes encoded as categorical values. As such, depending on the difference in encoded values for the predicted and the ground truth genotypes, the penalty can be severe. For example, if $0|0$, $0|1$, and $1|1$ are encoded as 0, 1, and 2 in genotypes and the ground truth for a given genotype is $0|0$, the model is punished moderately(severely) for predicting $0|1$($1|1$). Additionally, SCDA+ outperforms AE in most comparisons, indicating the effectiveness of our proposed training procedure.

Extensive structural variation datasets: In this experiment, we focus on R^2 between the predicted and ground truth genotypes as R^2 was the most discriminating metric for comparing the performance in imputing SVs. For estimating R^2, predictions are converted into categorical values, e.g., j**0**, 0j1, and 1j1 are encoded as 0, 1, and 2. Any discrepancy between the model's prediction and the ground truth leads to a substantial penalty on the correlation, enabling us to see di erences more clearly. STICI consistently outperformed Beagle5.4 and Minimac4 across various SV types, often by a noticeable margin. The underlying cause of this observation is the lack of high LD in this dataset (Figure 5) and fundamental di erences between HMMs and the Transformer model. In HMMs, information propagation between two distant variants occurs sequentially through intermediate sites. However, this mechanism falters when the LD block is sparse, leading to reduced performance. In contrast, STICI employs a direct variant-to-variant attention mechanism within each chunk without needing to model an intermediate site, which e ectively mitigates the limitations posed by a weak LD. Furthermore, the multi-head attention mechanism equips STICI to discern higher-order and complex patterns among variants, which appear to be crucial for better imputations in the absence of strong LD patterns. These capabilities highlight STICI's superiority in managing SV imputation challenges where traditional HMM-based approaches may be suboptimal. This is particularly the case for duplications (DUP) and insertions (INS) where STICI is able to attain a very high R^2 value. This observation matches our expectations since these two types of SVs are relatively challenging in genotype calling as well [35].

2.4 The relative performance of STICI for systematic missingness

In order to evaluate STICI against the competing methods for systematic missngness imputation, we curated four datasets that are missing approximately 90% of the variants in the test set. The rst dataset contains In nium Omni 2.5 BeadChip microarray dataset on human chromosome 22 (12,725 variants) as the test set and WGS genotypes from 1000 genomes project of the same region (99,314 variants) as the reference panel. We used the same individuals as [24] (100 samples from various populations) for the test set and the rest (2,404) for the reference panel. The second dataset was generated using stdpopsim [36] using msprime simulation engine [37]. There were 45,000 samples with 30,720 variants on human chromosome 19 in the reference panel and 5,000 samples with 3,044 variants for the same region in the test set. The third dataset contains 5,147 samples on a selected region on rat [38, 39] chromosome 20 with 61,440 variants as the reference panel, and 1000 samples with 6,140 variants scattered thorough the reference panel variants. The fourth dataset consists of 2,258 reference samples and 55,255 variants on Sasso chicken [40] chromosome 20 and 100 test samples 5,488 variants selected among the reference variants. More details about these datasets is provided in subsection 4.1.

We used accuracy, Impute info score (INFO score) [41], and Minimac R^2 [15] as evaluation metrics. The reason we included additional metrics for these experiments was that we would like to utilize multiple metrics so that the evaluation of model performance is more comprehensive and less biased. For example, accuracy is not considered a good metric for highly imbalanced data. In this case, accuracy for the variants with rare alleles is misleading because a method that always predicts the majority allele can retain a high accuracy. INFO score indicates the certainty of a model for alternative allele dosage prediction. In an imputation pipeline, INFO score is used to discard unreliable predictions. Minimac R^2 (Supplementary Equation 7) measures the squared correlation between imputed variants and the ground truth, and is useful for quality control and maintaining statistical power of genomic association studies.

During these experiments we noticed that the original implementation of STICI is not as accurate as classical models for rare alleles. To alleviate this problem, we developed a variant of STICI, namely STICI-R^2, that used Minimac R^2 as a new loss term to be used to improve imputation (more details in 4.2.6). We trained STICI and STICI- R^2, respectively, using a random MaskR (per sample seen in each training iteration) between 0.85 and 0.95. In other words, in each epoch the model would see various MaskRs in the aforementioned range for di erent samples. We nd this masking strategy more useful because in real human data, di erent segments had di erent MissRs, but the average was around 0.9 MissR.

The experimental results are presented in Figure 4, where each row is dedicated to the results of one dataset (experiment) and columns show accuracy, INFO score, and Minimac R^2 from left to right, respectively. While STICI R^2 achieved high accuracy ($\geq$ 0:95), high info score ($\geq$ 0:96), and high R2 ($>$ 0:48), some other methods produced almost perfect results for the simulated human data unlike that seen for the real data. One possibility for this performance di erence is that classical methods have

Fig. 3 Comparison sporadic imputation results of competing methods across SV types. Average R^2 of ground-truth genotypes in the test sets and respective predictions over 3-fold cross-validations on chromosomes 6, 10, 16, and 20. The experiments are performed on each chromosome separately, and the results are averaged over chromosomes and folds. Vertical lines indicate standard deviations. The improvement plot shows R^2 score difference between STICI and the best of other methods, normalized by the best R^2 scores for each SV type. We only report biallelic imputation results for SHAPEIT5 because we faced issues with imputing normalized multi-allelic variants using this software.

been fine-tuned over the years to accurately impute human genomes, unlike STICI. This prompted us to investigate the performance of all the methods on non-human genotype data, including data set from another mammal (rat) and a bird (Sasso chicken). Now, STICI-R^2 performs better than other methods, especially for imputing rare alleles. This may suggest that fine-tuning of classical models has increased their accuracy for human variants. We also recorded the time taken to impute for the competing methods. These timings can be found in theSupplementary Tables 10 & 11

3 Discussion

More accurate genotype imputation will improve the performance of downstream functional and biomedical genomic studies. Scientists frequently need to employ multiple tools, adapted based on the degree of missingness and types of variants missing, within individual pipelines to carry out genotype imputation. To address this problem, we have presented STICI, a masked DL framework, which appears to be one of the first uses of transformer architecture in imputing genomic data. While STICI is currently limited in a few ways, we believe that it represents a step towards developing a unified approach for successfully imputing missing values for a range of datasets, from small to large amount of missingness, as well as SNVs and SVs. We explored STICI's performance for a range of masking rates (training) and missing rates (application). Our experiments revealed that a single STICI model, trained with a masking rate

of 0.5, could be applied for imputing sporadic missingness of SNVs and SVs, while for systematic missingness which generally constitutes higher missing rates (above 80%), we can employ previously trained models using a masking rate similar to the missing rate in the test set. STICI's performance in imputing SNVs and SVs was comparable to many other methods and approaches for SNVs and SVs found in low and high LD regions. That is, STICI is capable of e ectively capturing short and long-range correlations among di erent genetic variants for genotype imputation.

STICI also performed well in imputing values that were missing systematically (Figure 1.b and 2.4). Furthermore, STICI o ers two additional advantages. Firstly, it can be applied directly to resequencing datasets from any species because. Unlike classical HMMs based on Li and Stephens model [42], a transformer model as implemented in STICI does not need hard-coded or parametric assumptions about underlying characteristics of the genomic data, such as mutation rate and density of LD blocks, and captures inherent patterns automatically. This was con rmed in experiments with chicken and rat datasets in subsection 2.4. Secondly, an STICI model needs to be trained once to impute sporadic and systematic missingness rapidly and accurately. Therefore, we expect the STICI framework to spur the next generation of approaches to advance generalized and e cient imputing, which could serve as online imputation servers because of the low computational burden of imputation after training.

Currently, training an STICI model is still resource- and time-demanding as transformers are computationally intensive. One of our development plans for STICI is to address these issues to develop lighter-weight models for genotype imputation. Moreover, the transformer blocks utilized in STICI are known to require a large number of samples to achieve optimal performance. Therefore, we expect STICI's performance to greatly improve even for variants in low LD regions as the number of samples is increased due to the release of new cohorts and data sources. For example, we plan to explore the potentials of improving STICI utilizing current and emerging genotypes for an increasing number of individuals in projects like UK Biobank [43], the Trans-Omics for Precision Medicine (TOPMed) [44], and All of Us [45].

4 Methods

In this section, we rst introduce the datasets we used in this study and discuss their characteristics. This is followed by the architectural design of STICI and the procedure for model training.

4.1 Data

We used eight datasets from four sequencing projects [10, 22, 38{40] and simulated a human genotype dataset using stdpopsim [36] with msprime simulation engine [37] to ne-tune and benchmark STICI against the baselines. The 1000 Genomes project datasets are pre-phased using SHAPEIT2. Thus, the phasing information of the test sets is propagated to the training sets, but the process is identical for all the methods so it will not bias the results in favor of any imputation method. Scikit-allel package [46] is employed to compute LD and MAF for the datasets. In the sporadic missingness experiments, we use 3-fold cross-validation to assess the performance of the methods. In a 3-fold cross-validation, each dataset is separated into three distinct partitions where there is no sample overlap. Each time, one of the partitions is used as the test set while the remaining two partitions are used for the training process. For the DL models, a validation set is selected from the training samples for early stopping. To ensure that the same training/validation/test set is used across di erent methods, we used xed random seeds for splitting the data into folds and introducing missingness into the test sets. For systematic missingness experiments, we selected a xed number of individuals as the test set. We used python scripts to ensure that no parental strands (haploids) are shared among the training and test sets in both sporadic and systematic experiments.

For sporadic missingness imputation, we used three schemes to introduce sporadic missingness: random selection, MAF-distributed selection, and LD-distributed selection. For the latter two, we computed MAF and LD on the whole data, as depicted in Figure 5, and selected the missing SNVs/SVs for the test set evaluation proportional to those. That is, if 10% of the variants have an MAF in the range of [0.2, 0.3), we selected 10% of the missing values from these speci c variants. This approach ensures that the missing data are imputed based on the distribution of MAF or LD in the data, providing a representative imputation strategy. Regardless of the scheme, we used xed random seeds per sample to decide the

missing genotypes. Therefore, the missing genotypes across the test samples are not identical. However, the coordinates of missing values are identical across the competing methods.

For systematic missingness imputation datasets, excluding Omni 2.5 BeadChip microarray imputation, we decided on a missing rate and randomly selected and removed the variants inthe following prede ned MAF bins:

$$[0{:}01; 0{:}05; 0{:}1; 0{:}2; 0{:}3; 0{:}4; 0{:}5]$$

The characteristics of the datasets we used in our experiments are as follows:

4.1.1 HLA dataset

This dataset contains human leukocyte antigen genotypes, covering a 3 Mbp region at chromosome 6p21.31 and sitting at a major histocompatibility complex (MHC) region. HLA region regulates the immune system in humans [47]. It is highly polymorphic and heterogeneous among individuals; i.e., it harbors various alleles, enabling the adaptive immune system to be ne-tuned [48]. In this study, we used the genotypes of this region, obtained from phase 3 of the 1000 Genomes Project [10], which contained 7161 unique genetic variants for 2,504 individuals from ve super-populations across the world: American (AMR), East Asian (EAS), European (EUR), South Asian (SAS), and African (AFR). The majority of SNVs in this dataset exhibit maximum LD values in the range of $[0{:}9; 1{:}0]$. We used this dataset for our masking study and ne-tuning the hyper-parameters of SCDA, AE, and STICI.

4.1.2 Yeast dataset

The second dataset is the comprehensively assayed yeast dataset [22], representing a simple genetic background and high correlation among genotypes. This dataset contains 4,390 genotyped pro les for 28,220 genetic variants. The samples were obtained by sequencing crosses between two strains of yeast, namely an isolate from a vineyard (RM) and a popular laboratory strain (BY). In the original dataset, the data is encoded as -1/1 for BY/RM, which are mapped to 0/1 in our code, respectively, before one-hot encoding.

4.1.3 Chromosome 22 datasets

We used structural variation data from the 1000 Genomes Project in two settings. In the rst, we only selected deletions (DEL), excluding ALU/SVA/LINE1 deletions, among all SVs. This resulted in 573 positions harboring bi-allelic events in the dataset. In the second, a total of 848 SVs including, but not limited to deletions, insertions, duplications, inversions (INV), and copy number variations (CNV) in chromosome 22 are selected. As shown in Figure 5 b & c, the majority of SVs in chromosome 22 exhibit a low LD, rendering these datasets challenging for imputation compared to SNVs. According to Figure 5.d, deletions cover a wide range of LD among them and other SVs, making them a good target for a separate bi-allelic dataset.

4.1.4 Extensive structural variation datasets

In the concluding experiment, we undertook a thorough investigation of SV imputation using the human 1000 Genomes Project, selecting 4187, 3126, 2062, and 1569 SVs located in chromosomes 6, 10, 16, and 20, respectively. This selection strategy was informed by the aim to encompass chromosomes of di erent lengths, providing a representative cross-section of the genome. This diverse chromosome selection allows for a broader understanding of the genomic distribution and characteristics of SVs, facilitating a more nuanced analysis of their presence and impact across di erent regions of the human genome. These SVs include deletions, duplications, insertions, inversions, and copy number variations. Among these SVs, 469 of them are multi-allelic (CNVs). For each model, we train on and impute each chromosome separately, and take the average of the results over folds, chromosomes, and SV type. The distributions of SVs in these chromosomes in terms of MAF and LD are presented in Figures 5 e, f, g, and h, indicating low LD and diverse MAF in general for the mentioned SV datasets.

4.1.5 Systematic missingness imputation

For the rst dataset (Figure 4 a & b & c) we used the SNVs ($MAF > 0{:}01$) in chromosome 22 dataset from human 1000 genomes project, and used PLINK2 [49] and bcftools [50] to preprocess the data and

convert multi-allelic events, and selected the first 99,314 variants in this chromosome. Following the instructions for data collection in [24] and a script provided by the authors, we created a microarray dataset using Illumina Omni 2.5 BeadChip manifest for chromosome 22 containing 12,725 in the same region. We followed [24] for selecting the exact individuals for the microarray data (test set), and the rest for the reference panel.

The second dataset (Figure 4d & e & f) was generated using stdpopsim [36] and msprime simulation engine [37] using a reference panel and demographic model (four population out-of-Africa history) integrated into stdpopsim package [51–55]. This dataset contained simulated CEU population samples with a minimum MAF of 0.01 on chromosome 19. We selected 45,000 samples as the reference panel and 5,000 samples with unique parental strands (haploids) for the test set, and shortlisted the first 30,720 variants for the reference panel. Out of 30,720 variants present in the reference panel, 90% of them in each MAF bin (described at 4.1) were removed, leaving 3,044 variants in the test samples. It is worth mentioning that there were shared parental strands (haploids) in the reference panel but the parental strands in the test set were all unique within the test set and among the test set and the reference panel.

The rat dataset [38, 39](Figure 4g & h & i) contains 5,147 outbred samples from more than 10 projects on a selected region at rat chromosome 20 with 61,440 variants as the reference panel, and 1000 samples with 6,140 variants scattered thorough the reference panel variants. To preserve parental strand uniqueness, we used variants with a minimum MAF of 0.001.

The Sasso chicken dataset [40] (Figure 4j & k & l) was already pre-processed by the curators and non-biallelic SNVs and SNVs with MAF lower than 0.02 were removed. This dataset constitutes of 2,258 pre-processed samples and 55,255 variants on chicken chromosome 20 and 100 test samples with 5,488 variants selected among the reference variants. The test variants were obtained by randomly removing 90% of the reference panel variants in each MAF bin described earlier.

We used python scripts, PLINK2 [49] and bcftools [50] to pre-process the data and we used SHAPEIT5 [17] to impute the sporadic missing data (pre-phasing) for the rat and chicken datasets.

In previous studies, the training data is masked using different rates to match the test set, e.g., [3, 14]. In our experiments, we observed improved performance of the model with 50% dynamic and random masking of the variants in the training data. So we trained the DL model once and reused it multiple times for sporadic missingness ($MissR < 0.5$). For higher MissRs, we can train multiple models for different MissR bins (e.g., $0.8 < MissR < 0.9$ and $0.9 < MissR < 0.99$) and use the proper saved model based on the missing rate in the target data. Notably, this masking is similar to the masking performed in modern large language models. The benefit of such a technique in genomic data imputation is the notable reduction in the inference (imputation) times when compared to the fastest traditional methods. Consequently, a DL model trained in this manner becomes particularly advantageous for deployment on imputation servers, where re-training needs to be avoided for quick and efficient processing.

Another improvement we achieved was by representing phased diploids into haploids, followed by one-hot encoding. That is, instead of feeding (one-hot encoded) phased diploids to the models, we fed them haploids. This idea is proposed in [27], but there is no discussion about the merits of this procedure. We surmised that predicting haploids would be easier because mutations in paternal and maternal haploids are independent of each other. In the output, diploid genotypes were reconstructed by combining corresponding haploids together.

4.2 STICI architecture

Split-Transformer Impute is an extended transformer model [29] specifically tailored for genotype imputation. STICI models do not require any additional information provided by a reference panel, except for the genotypes and their relative positions. This makes STICI adaptable to any genotype data and allows it to be applied to a wider range of datasets with less effort and fewer preparations. Moreover, although here we focus on sporadic missingness, once STICI is trained on a dataset, it can predict both sporadic missingness and systematic missingness in genotype data as long as the target variants are a subset of the training variants. An overview of STICI is presented in Figure 6. We implemented STICI and the rest of the DL models using Tensorflow framework [56] in Python. In order to train the models, we used tensor processing units (TPU) provided by the Google Colaboratory platform, but a GPU implementation of STICI is available as well. A learning rate scheduler and early stopping are employed in order to reduce the loss and training duration.

4.2.1 Cat-Embedding

One important part of STICI is categorical embedding (Figure 6.b), termed as Cat-Embedding, which enables it to learn embedding representation per allele in each position. For the imputation task, we consider missing values as another allele that is equivalent to special tokens in natural language processing. The corresponding vector for each allele is added to the respective positional variant embedding vector to generate the final embedding. The idea is similar to a natural language processing embedding layer that accepts word indices, except that Cat-Embedding accepts one-hot encoded data.

4.2.2 Splitting

While the multi-headed attention in a transformer offers significant advantages, a major drawback is quadratic memory cost for computations that becomes important in genomic analysis, since the number of variants in a sample is normally in the thousands. In genotypes, the majority of interactions are local [57]. Therefore, it is of great importance to limit the scope of attention to save computational resources. To do so, we split the variants into chunks (vertical partitioning). The chunk size and overlap size are employed in a comparable manner in Minimac4.1.4 and analogous software applications. In order to prevent loss of imputation accuracy at chunk borders, we include flanking variants from neighboring chunks and discard them after applying self-attention to get the original variants in the chunk. Though the average LD block size in the dataset can be used to decide the size of overlap, we do not use LD blocks directly to decide the chunk size in the current version.

Each chunk passes through a dedicated branch inside the model, leading to increased imputation quality. Ideally, having a vast number of samples allows training a single model with attention across the whole genome. However, when the number of samples is not enough, the model is left with untrained parameters, resulting in poor performance. Hence, chunking regulates the number of parameters. In a vanilla transformer, the cost of computing global attention is quadratic with respect to the number of SNVs (m^2); however, the amount is lowered to $(m/w)\ (w + o)^2 = mw$ in STICI, considering that the overlaps of chunks are negligible compared to the chunk size. For instance, for $m = 10^4$ and a chunk size of 10^3, STICI uses 10 times less memory for attention computations compared to a vanilla transformer.

4.2.3 Attention

The attention blocks are implemented similarly to those of other transformers, such as self-attention blocks in Vision Transformer (ViT) [58]. There is a difference between the first and second attention blocks in the branches. The first block is a self-attention block, meaning that the query, key, and value of the attention layer are the same. The output of multi-head attention in Tensorflow has the same dimensions as the query. By excluding the neighboring variants of a chunk from the query and only including them in the key and value, we involve them in the attention mechanism and, at the same time, shrink the output of a chunk to the target size (chunk size without counting flanking/overlap variants) after applying multi-headed attention. In the second block, the query is the output of the previous layer, while the key and value are the outputs of the first self-attention block. This skip connection considerably affects the overall performance of the model.

4.2.4 Convolutional block

Convolutional blocks, as illustrated in Figure 6.c, are also crucial components of STICI. Through empirical studies, we found that using two parallel convolutional branches with varying kernel sizes, similar to the Inception module [59], is the best trade-off between accuracy gain and increase in a number of model parameters, compared to using a single branch or more than two branches. Furthermore, a Depth-wise convolutional layer at the end of the block helps STICI extract local information without mixing channel information and substantially improves imputation accuracy.

4.2.5 Output formation

Finally, the outputs of all branches are concatenated to form the output, that is, either maternal or paternal haplotype in the case of 1000 Genomes Project datasets or the genotypes in the case of yeast. For the former, by assembling maternal and paternal haplotypes, we obtain imputed genotypes, and

the latter needs no further post-processing. Since genetic variations in parents are independent, directly encoding and imputing the genotypes in diploid life forms results in lower imputation accuracy compared to imputing their haplotypes. Hence we undergo extra steps to separate diplotypes into haplotypes in pre-processing, and combining respective predicted haplotypes into diplotypes in post-processing for the human, chicken, and rat datasets.

4.2.6 Loss function

For the loss function, we used a combination of Kullback{Leibler divergence (D_{KL}) and categorical cross-entropy (CCE), similar to the loss function of variational autoencoder [60], as follows:

$$\text{Loss}(y;\hat{y}) = (\quad)CCE(y;\hat{y}) + (1 \qquad)D(y\|\hat{y}); \tag{1}$$

where is the weight parameter. The rst term, representing categorical cross-entropy, and the second term, representing Kullback{Leibler divergence loss, are calculated as follows:

$$CCE(y;\hat{y}) = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\ \log\ p(y_{ij}) \tag{2}$$

$$D_{KL}(y\|\hat{y}) = \sum_{i=1}^{N} p(y_i)\frac{p(y_i)}{p(\hat{y}_i)} \tag{3}$$

We set to 0:5, meaning that STICI minimizes Equations 2, 3 equally. CCE captures reconstruction error between the input and the output, while D_{KL} measures asymmetric distance, with$y$ as the base, between their probability distributions. In our experiments, omitting any of these losses resulted in reduced model performance. Theoretically, KL-divergence and cross-entropy are related and using both might not seem to contribute to the performance of the model. However, adding KL-divergence to the loss term helps the model to retain the probability/dosage distribution of alleles per variant. In other words, while cross-entropy focuses on predicting the correct genotype, KL-divergence acts as a regularization factor and penalizes the model whenever the shape of the predicted probability distribution (allele probability/dosage) shows divergence from the ground truth. Moreover, the mathematical relation of D_{KL} and CCE can be summarized as follows:

$$D_{KL}(y\|\hat{y}) = CCE(y;\hat{y}) \quad CCE(y) \tag{4}$$

where $CCE(y)$ is the entropy of the ground truth. According to Equation 4 minimizing $D_{KL}(y\|\hat{y})$ is equivalent to minimizing $CCE(y;\hat{y})$ under the condition that the entropy of the ground truth remains constant. However, in deep learning models, data is typically processed in mini-batches. This means that the entropy of each mini-batch may not accurately represent the entropy of the entire ground truth. As a result, Equation 4 does not hold for the DL models in general.

We also used MinimacR^2 as an additional loss term for STICI-R^2. This loss is calculated for each variant/site in each sample as follows:

$$\text{Minimac } R^2 \text{ loss}_i = \frac{(\sum_{j=2}^{n_a} D_i^j \quad \hat{p})^2}{\hat{p}(1 \quad \hat{p})} \tag{5}$$

where $\hat{p}$ is the alternate allele frequency for the ground truth at variant i in the current batch, N_a is the number of alleles,D_i^j is the imputed allele probability at the i^{th} haplotype, and $\sum_{j=2}^{n_a} D_i^j$ is the sum of predicted probabilities of alternative alleles for the i^{th} site. Notably, adding this loss causes model to put more weight on the loss generated by rare alleles. This loss is sensitive to batch size and with an increase in batch size, this loss term chips away at other loss terms. We found out that using a batch size of 4 for training STICI- R^2 presents us with the best trade-o among accuracy andR^2 metrics.

4.3 Baseline models

In order to benchmark our model, we compare STICI to state-of-the-art imputation models capable of imputing sporadic missingness: SCDA [14], AE [3], Impute5 [18], SHAPEIT5 [17], Eagle2 [34] Beagle5.4 [19], and Minimac4.1.4 [15]. In [14], experimental results indicate that SCDA outperforms shallow ML

models for genotype imputation. Hence, we do not include shallow ML models in our benchmarking analyses. Additionally, in order to assess the contribution of Cat-Embedding, we replaced it with a convolution layer in STICI, named the resulting model STICI-NE, ne-tuned it, and applied it to the benchmark datasets. Lastly, we trained SCDA, in addition to STICI, using our proposed pre-processing and training procedure, and compared it to AE. Since AE and original SCDA are the same and only di er in pre-processing step (which results in AE outperforming SCDA), we believe that this comparison can demonstrate the e ectiveness of our proposed pre-processing and training procedure.

For SCDA and AE, hyper-parameter tuning information on the yeast dataset is present in the original papers. For SCDA, AE, and STICI, we conducted a grid search for optimal hyper-parameters on the HLA dataset using validation sets in a 3-fold cross-validation. We assessed the impact of these hyper-parameters on the performance of the models within the HLA dataset and applied these ndings to select suitable hyper-parameters for the yeast dataset in the case of STICI, and for the SV dataset across all four mentioned methods. The upper limit for the hyper-parameters was the resource limit of Google Colaboratory using Nvidia Titan IV GPU with 16 GB of RAM size for AE, and roughly the same limitation for TPU RAM size. Classical imputation tools, such as Minimac, do not require ne tuning for the experiments we run.

4.4 Experimental settings

The input to all DL models is one-hot encoded. While STICI can handle diploids, we found that the best performance was achieved when the inputs of the DL models were haplotypes, an analysis inspired by [27]. Therefore, for the HLA dataset and chromosome 22 datasets, we separated each diplotype into maternal and paternal haplotypes, fed them into the model, and reconstituted the resulting predictions for SCDA [14] and STICI. We continue using diplotypes as inputs for AE [3] since it is an improved version of SCDA in which the training process was modi ed, and we wanted to keep it intact. By doing so, we also compare the improvement in AE to our implementation of SCDA, called SCDA+, in which we use proposed pre-processing in conjunction with the changes to the training process as a contribution. The yeast dataset contains haplotypes, so there is no need for the aforementioned extra steps.

In this study, to evaluate the imputation power of the models, multiple evaluation metrics are used including imputation accuracy, imputation quality score (IQS) [61], weighted F 1-score, correlation between imputed and real genotypes in terms of R^2 [62], Minimac R^2 [15], and INFO score[41]. Accuracy and weighted F 1-score are calculated only for positions with missing genotypes and for these metrics, heterozygous genotypes are encoded di erently; i.e.0j1 and 1j0 are encoded to two di erent categorical values. IQS adjusts the chance concordance between predicted and the ground truth SNVs and is de ned for bi-allelic events. Therefore, IQS cannot be calculated for any SV in chromosome 22. R^2 is the squared Pearson correlation coe cient between the imputed genotypes and the true genotypes at a speci c locus. Minimac R^2 measures the mean squared error between the predicted alternative allele dosage and the ground truth dosage. INFO score is primarily used for quality control and indicates the quality of imputation. The de nition of these metrics is provided in the Metrics section of the Supplement. For our experiments, we used python implementation of MinimacR^2 and INFO score provided in the Github reposity of [24].

Data availability

All data used in this study are publicly available. The yeast dataset can be found as the Supplementary Data 5 at https://www.nature.com/articles/ncomms9712, the rest of datasets for sporadic missingness imputation are extracted from the 1000 Genomes Project phase 3 dataset available at http://ftp.1000genomes.ebi.ac.uk/vol1/ftp/release/20130502/. Instructions on how to prepare the data for human chromosome 22 systematic missingness can be found in https://github.com/kanamekojima/rnnimp. The Rat dataset can be accessed at https://library.ucsd.edu/dc/ using accession code bb15123938and the chicken genotype data can be found in https://datashare.ed.ac.uk/handle/10283/8761.

Code availability

The source code of STICI is publicly available on GitHub (https://github.com/ErfanMowlaei/STICI).

5 Acknowledgements

This work is partially supported by the US National Science Foundation (DBI 1750632) and the National Institutes of Health (GM-0126567-03). This research includes calculations carried out on HPC resources supported in part by the National Science Foundation through major research instrumentation grant number 1625061 and by the US Army Research Laboratory under contract number W911NF-16-2-0189. We appreciate the suggestions provided by Dr. Francisco McGee, John Allard, Rohan Alibutud, and Vahid Mahzoon that helped us improve the model performance and design the experiments. Additionally, we would like to thank Dr. Kaname Kojima for helping us obtain the data for the Missing variant experiment and Emily Thyrum for proofreading the manuscript.

6 Contributions

M.E.M. developed the method with the help of J.C., B.J., V.C, and X.S. and M.E.M. implemented the code. C.L. and O.J. prepared the datasets. M.E.M. and R.D. performed the experiments. M.E.M, S.K., C.L., O.J., and X.S. design and/or conducted data analysis. M.E.M, C.L., S.K., T.R.R., V.C, and X.S. wrote the manuscript. All the authors read and approved the submitted manuscript.

Ethics declarations

Not applicable.

Competing interests. The authors declare that they have no competing interests.

Declarations

Not applicable.

References

[1] Lewis, C.M., Vassos, E.: Polygenic risk scores: from research tools to clinical instruments. Genome medicine 12(1), 1{11 (2020)

[2] Torkamaneh, D., Belzile, F.: Accurate imputation of untyped variants from deep sequencing data. Deep Sequencing Data Analysis, 271{281 (2021)

[3] Song, M., Greenbaum, J., Luttrell IV, J., Zhou, W., Wu, C., Luo, Z., Qiu, C., Zhao, L.J., Su, K.-J., Tian, Q., et al.: An autoencoder-based deep learning method for genotype imputation. Frontiers in Arti cial Intelligence 5 (2022)

[4] Das, S., Abecasis, G.R., Browning, B.L.: Genotype imputation from large reference panels. Annu Rev Genomics Hum Genet 19(1), 73{96 (2018)

[5] Gra elman, J., Nelson, S., Gogarten, S., Weir, B.: Exact inference for hardy-weinberg proportions with missing genotypes: Single and multiple imputation. G3: Genes, Genomes, Genetics 5(11), 2365{2373 (2015)

[6] Wigginton, J.E., Cutler, D.J., Abecasis, G.R.: A note on exact tests of hardy-weinberg equilibrium. The American Journal of Human Genetics 76(5), 887{893 (2005)

[7] Pei, Y.-F., Li, J., Zhang, L., Papasian, C.J., Deng, H.-W.: Analyses and comparison of accuracy of di erent genotype imputation methods. PloS one 3(10), 3551 (2008)

[8] Auer, P.L., Wang, G., Project, N.E.S., Leal, S.M.: Testing for rare variant associations in the presence of missing data. Genetic epidemiology 37(6), 529{538 (2013)

[9] Chiang, C., Scott, A.J., Davis, J.R., Tsang, E.K., Li, X., Kim, Y., Hadzic, T., Damani, F.N., Ganel, L., Consortium, G., et al.: The impact of structural variation on human gene expression. Nature genetics 49(5), 692–699 (2017)

[10] Consortium, .G.P., et al.: A global reference for human genetic variation. Nature 526(7571), 68 (2015)

[11] Liu, Z., Roberts, R., Mercer, T.R., Xu, J., Sedlazeck, F.J., Tong, W.: Towards accurate and reliable resolution of structural variants for clinical diagnosis. Genome biology 23(1), 68 (2022)

[12] Bartlett, J.W., Seaman, S.R., White, I.R., Carpenter, J.R., Initiative*, A.D.N.: Multiple imputation of covariates by fully conditional specification: accommodating the substantive model. Statistical methods in medical research 24(4), 462–487 (2015)

[13] Song, M., Greenbaum, J., Luttrell IV, J., Zhou, W., Wu, C., Shen, H., Gong, P., Zhang, C., Deng, H.-W.: A review of integrative imputation for multi-omics datasets. Frontiers in genetics 11, 570255 (2020)

[14] Chen, J., Shi, X.: Sparse convolutional denoising autoencoders for genotype imputation. Genes 10(9), 652 (2019)

[15] Das, S., Forer, L., Schönherr, S., Sidore, C., Locke, A.E., Kwong, A., Vrieze, S.I., Chew, E.Y., Levy, S., McGue, M., et al.: Next-generation genotype imputation service and methods. Nature genetics 48(10), 1284–1287 (2016)

[16] Li, Y., Willer, C.J., Ding, J., Scheet, P., Abecasis, G.R.: Mach: using sequence and genotype data to estimate haplotypes and unobserved genotypes. Genetic epidemiology 34(8), 816–834 (2010)

[17] Hofmeister, R.J., Ribeiro, D.M., Rubinacci, S., Delaneau, O.: Accurate rare variant phasing of whole-genome and whole-exome sequencing data in the uk biobank. Nature Genetics 55(7), 1243–1249 (2023)

[18] Rubinacci, S., Delaneau, O., Marchini, J.: Genotype imputation using the positional burrows wheeler transform. PLoS genetics 16(11), 1009049 (2020)

[19] Browning, B.L., Zhou, Y., Browning, S.R.: A one-penny imputed genome from next-generation reference panels. The American Journal of Human Genetics 103(3), 338–348 (2018)

[20] Scheet, P., Stephens, M.: A fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypes and haplotypic phase. The American Journal of Human Genetics 78(4), 629–644 (2006)

[21] Rubinacci, S., Hofmeister, R.J., Mota, B., Delaneau, O.: Imputation of low-coverage sequencing data from 150,119 uk biobank genomes. Nature Genetics 55(7), 1088–1090 (2023)

[22] Bloom, J.S., Kotenko, I., Sadhu, M.J., Treusch, S., Albert, F.W., Kruglyak, L.: Genetic interactions contribute less than additive effects to quantitative trait variation in yeast. Nature communications 6(1), 1–6 (2015)

[23] Dias, R., Evans, D., Chen, S.-F., Chen, K.-Y., Loguercio, S., Chan, L., Torkamani, A.: Rapid, reference-free human genotype imputation with denoising autoencoders. Elife 11, 75600 (2022)

[24] Kojima, K., Tadaka, S., Katsuoka, F., Tamiya, G., Yamamoto, M., Kinoshita, K.: A genotype imputation method for de-identified haplotype reference information by using recurrent neural network. PLoS Computational Biology 16(10), 1008207 (2020)

[25] Kojima, K., Tadaka, S., Okamura, Y., Kinoshita, K.: Two-stage strategy using denoising autoencoders for robust reference-free genotype imputation with missing input genotypes. Journal of Human

Genetics, 1{8 (2024)

[26] Chi Duong, V., Minh Vu, G., Khac Nguyen, T., Tran The Nguyen, H., Luong Pham, T., S. Vo, N., Hong Hoang, T.: A rapid and reference-free imputation method for low-cost genotyping platforms. Scienti c Reports 13(1), 23083 (2023)

[27] Naito, T., Suzuki, K., Hirata, J., Kamatani, Y., Matsuda, K., Toda, T., Okada, Y.: A deep learning method for hla imputation and trans-ethnic mhc ne-mapping of type 1 diabetes. Nature communications 12(1), 1{14 (2021)

[28] Tanaka, K., Kato, K., Nonaka, N., Seita, J.: E cient hla imputation from sequential snps data by transformer. arXiv preprint arXiv:2211.06430 (2022)

[29] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems 30 (2017)

[30] Desimone, R., Duncan, J.: Neural mechanisms of selective visual attention. Annual review of neuroscience 18(1), 193{222 (1995)

[31] Cho, K., Courville, A., Bengio, Y.: Describing multimedia content using attention-based encoder-decoder networks. IEEE Transactions on Multimedia 17(11), 1875{1886 (2015)

[32] Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Zdek, A., Potapenko, A., et al.: Highly accurate protein structure prediction with alphafold. Nature 596(7873), 583{589 (2021)

[33] Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., et al.: Evolutionary-scale prediction of atomic-level protein structure with a language model. Science 379(6637), 1123{1130 (2023)

[34] Loh, P.-R., Danecek, P., Palamara, P.F., Fuchsberger, C., A Reshef, Y., K Finucane, H., Schoenherr, S., Forer, L., McCarthy, S., Abecasis, G.R., et al.: Reference-based phasing using the haplotype reference consortium panel. Nature genetics 48(11), 1443{1448 (2016)

[35] Sudmant, P.H., Rausch, T., Gardner, E.J., Handsaker, R.E., Abyzov, A., Huddleston, J., Zhang, Y., Ye, K., Jun, G., Hsi-Yang Fritz, M., et al.: An integrated map of structural variation in 2,504 human genomes. Nature 526(7571), 75{81 (2015)

[36] Adrion, J.R., Cole, C.B., Dukler, N., Galloway, J.G., Gladstein, A.L., Gower, G., Kyriazis, C.C., Ragsdale, A.P., Tsambos, G., Baumdicker, F., et al.: A community-maintained standard library of population genetic models. elife 9, 54967 (2020)

[37] Kelleher, J., Etheridge, A.M., McVean, G.: E cient coalescent simulation and genealogical analysis for large sample sizes. PLoS computational biology 12(5), 1004842 (2016)

[38] Gunturkun, M.H., Wang, T., Chitre, A.S., Garcia Martinez, A., Holl, K., St. Pierre, C., Bimschleger, H., Gao, J., Cheng, R., Polesskaya, O., et al.: Genome-wide association study on three behaviors tested in an open eld in heterogeneous stock rats identi es multiple loci implicated in psychiatric disorders. Frontiers in Psychiatry 13, 790566 (2022)

[39] Gileta, A.F., Gao, J., Chitre, A.S., Bimschleger, H.V., St. Pierre, C.L., Gopalakrishnan, S., Palmer, A.A.: Adapting genotyping-by-sequencing and variant calling for heterogeneous stock rats. G3: Genes, Genomes, Genetics 10(7), 2195{2205 (2020)

[40] Morris, K.M., Sutton, K., Nedi, M.G., Sanchez Molano, E., Solomon, B., Esatu, W., Alemayehu, T.D., Vervelde, L., Psi di, A., Hanotte, O., Banos, G.: Genotype data of Sasso chicken. University of Edinburgh. Centre For Tropical Livestock Genetics and Health (2024). https://doi.org/10.7488/

ds/7718 . https://doi.org/10.7488/ds/7718

[41] Howie, B.N., Donnelly, P., Marchini, J.: A flexible and accurate genotype imputation method for the next generation of genome-wide association studies. PLoS genetics **5**(6), 1000529 (2009)

[42] Li, N., Stephens, M.: Modeling linkage disequilibrium and identifying recombination hotspots using single-nucleotide polymorphism data. Genetics **165**(4), 2213–2233 (2003)

[43] Sudlow, C., Gallacher, J., Allen, N., Beral, V., Burton, P., Danesh, J., Downey, P., Elliott, P., Green, J., Landray, M., *et al.*: Uk biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. PLoS medicine **12**(3), 1001779 (2015)

[44] Taliun, D., Harris, D.N., Kessler, M.D., Carlson, J., Szpiech, Z.A., Torres, R., Taliun, S.A.G., Corvelo, A., Gogarten, S.M., Kang, H.M., *et al.*: Sequencing of 53,831 diverse genomes from the nhlbi topmed program. Nature **590**(7845), 290–299 (2021)

[45] Us Research Program Investigators, A.: The "all of us" research program. New England Journal of Medicine **381**(7), 668–676 (2019)

[46] Miles, A., bot, R., M., Ralph, P., Harding, N., Pisupati, R., Rae, S., Millar, T.: Cggh/scikit-allel: V1.3.3. https://doi.org/10.5281/zenodo.4759368 . https://doi.org/10.5281/zenodo.4759368

[47] Hillert, J.: Human leukocyte antigen studies in multiple sclerosis. Annals of Neurology: Official Journal of the American Neurological Association and the Child Neurology Society **36**(S1), 15–17 (1994)

[48] Terasaki, P.I., Cai, J.: Human leukocyte antigen antibodies and chronic rejection: from association to causation. Transplantation **86**(3), 377–383 (2008)

[49] Chang, C.C., Chow, C.C., Tellier, L.C., Vattikuti, S., Purcell, S.M., Lee, J.J.: Second-generation plink: rising to the challenge of larger and richer datasets. Gigascience **4**(1), 13742–015 (2015)

[50] Li, H.: A statistical framework for snp calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. Bioinformatics **27**(21), 2987–2993 (2011)

[51] Takahata, N.: Allelic genealogy and human evolution. Molecular biology and evolution **10**(1), 2–22 (1993)

[52] Tremblay, M., Vézina, H.: New estimates of intergenerational time intervals for the calculation of age and origins of mutations. The American Journal of Human Genetics **66**(2), 651–658 (2000)

[53] Spence, J.P., Song, Y.S.: Inference and analysis of population-specific fine-scale recombination maps across 26 diverse human populations. Science Advances **5**(10), 9206 (2019)

[54] Consortium, I.H., *et al.*: A second generation human haplotype map of over 3.1 million snps. Nature **449**(7164), 851 (2007)

[55] Jouganous, J., Long, W., Ragsdale, A.P., Gravel, S.: Inferring the joint demographic history of multiple populations: beyond the diffusion approximation. Genetics **206**(3), 1549–1567 (2017)

[56] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., et al.: Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467 (2016)

[57] Weir, B.: Linkage disequilibrium and association mapping. Annual review of genomics and human genetics **9**(1), 129–142 (2008)

[58] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)

[59] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)

[60] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)

[61] Lin, P., Hartz, S.M., Zhang, Z., Saccone, S.F., Wang, J., Tischfield, J.A., Edenberg, H.J., Kramer, J.R., M. Goate, A., Bierut, L.J., *et al.*: A new statistic to evaluate imputation reliability. PloS one **5**(3), 9697 (2010)

[62] Deng, T., Zhang, P., Garrick, D., Gao, H., Wang, L., Zhao, F.: Comparison of genotype imputation for snp array and low-coverage whole-genome sequencing data. Frontiers in genetics **12**, 704118 (2022)

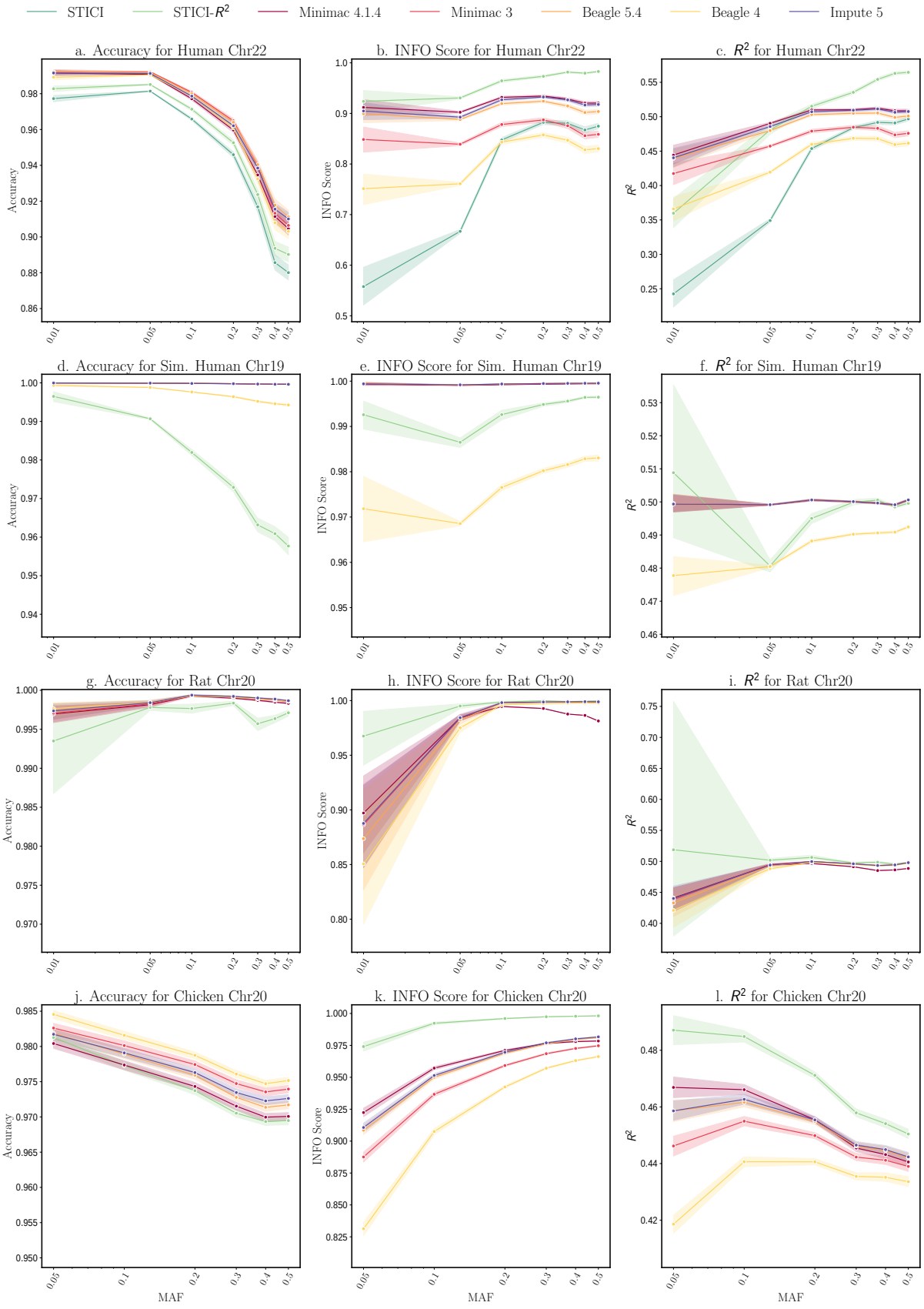

Fig. 4 Systematic missingness imputation results across di erent datasets. The results for each dataset is arranged in one row (human Chr22 in **a** & **b** & **c**, simulated human Chr19 in **d** & **e** & **f**, rat Chr20 in **g** & **h** & **i**, Sasso chicken Chr20 in **j** & **k** & **l**). The columns from left to right respectively contain accuracy, INFO score, and Minimac R^2 results. The lines show the average of the metrics while the bands around each line indicate 95% confidence interval.

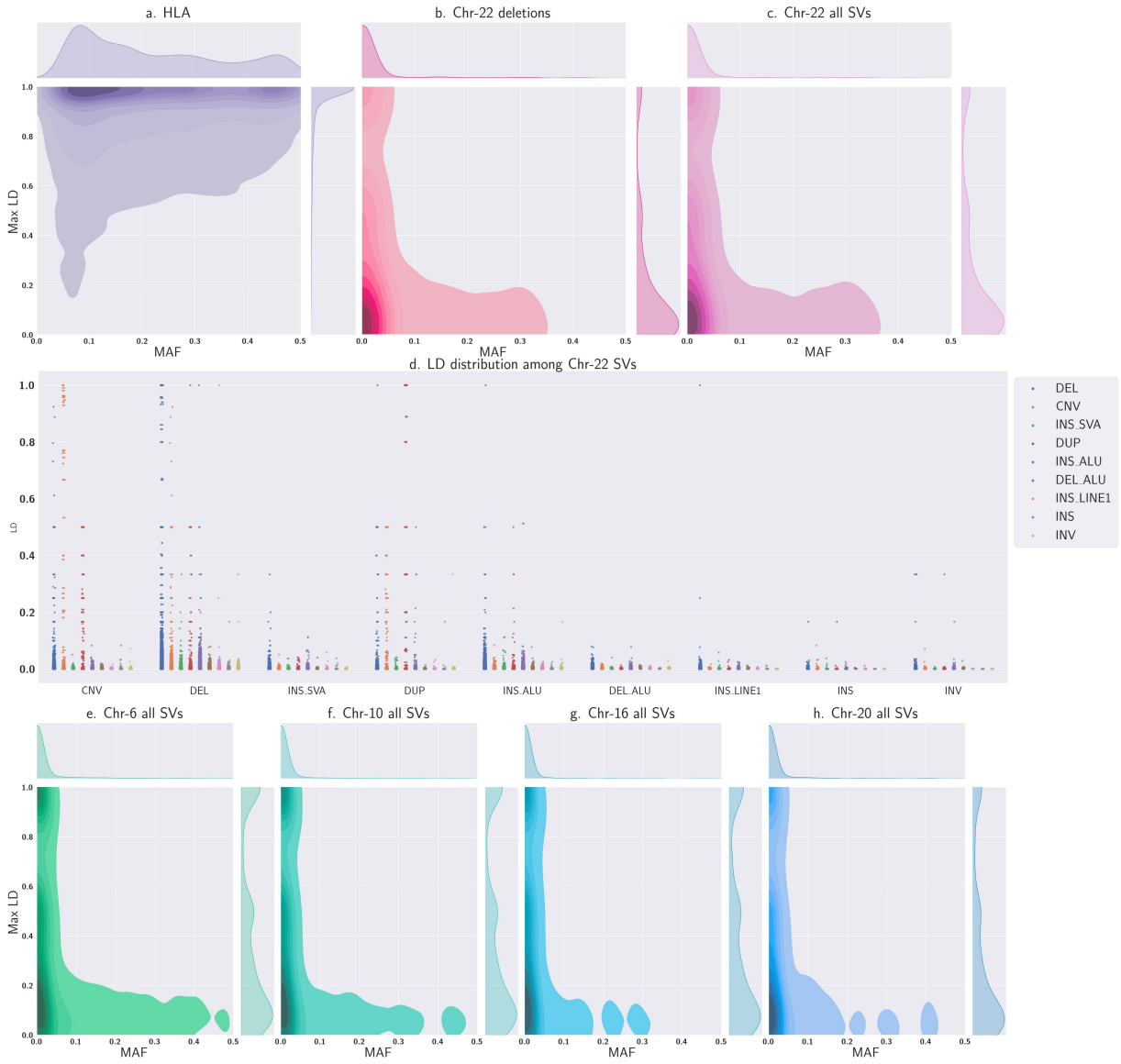

Fig. 5 MAF and LD distributions of benchmark datasets from the 1000 Genomes Project. MAF and maximum LD distributions are presented using kernel density estimation plots for SNVs and SVs in *a. HLA region on chromosome 6*, *b. deletions in chromosome 22*, *c. SVs in chromosome 22*, *e. SVs in chromosome 6*, *f. SVs in chromosome 10*, *g. SVs in chromosome 16*, and *h. SVs in chromosome 20*. Overall, SVs exhibit a low LD value, posing a significant challenge to imputation methods. Plot *d. LD among di erent SV types in chromosome 22* shows that structural events are commonly correlated with deletions. Furthermore, deletion, copy number variation, and duplication events appear in different ranges of LD, while the rest of the events are limited to $LD \leq 0.1$. Lastly, the majority of correlated SVs to deletions are of the same event, making deletions a good separate dataset for our experiment.