scientific reports

Check for updates

OPEN

Advanced analysis of retrotransposon variation in the human genome with nanopore sequencing using RetroInspector

Javier Cuenca-Guardiola[1], Belén de la Morena-Barrio[2], Javier Corral[2] & Jesualdo Tomás Fernández-Breis[1]✉

Transposable elements (TEs) make up 45% of the human genome, are a source of genetic variability difficult to detect, and involved in processes related to gene regulation and disease. Nanopore sequencing is recognized as one of the best technologies for detecting TEs; however, tools for analyzing of human TE insertions and deletions with nanopore-based data can be improved. RetroInspector is an easy to use, configurable Snakemake pipeline that performs detection, annotation, enrichment, and genotyping of TEs. RetroInspector requires the FASTQ files of the samples and the reference genome to start the identification and analysis of TEs. The user can also set the threshold for the number of supporting reads for the variant filtering. RetroInspector also allows users to compare the results of two samples. Different versions of the reference genome can be used and the presence of retrotransposition features can be annotated. RetroInspector has been run on three nanopore sequencing datasets and validated experimentally using proprietary and public data with over 80% precision.

Transposable elements (TEs) comprise approximately 45% of the human genome[1]. Part of this abundance stems from their ability to jump to different coordinates in the genome. Classifying TEs is a complex task due to several factors: first, similar to viruses, they do not share a common ancestor, which makes it difficult to construct a phylogeny; second, they evolve by a combination of different overlapping phenomena, such as truncated or nested insertions; third, the same group of TEs can originate from different events, as SINEs have[2].

In general, TEs can be classified based on their transposition mechanism into two groups: class I elements, or retrotransposons, which produce an RNA intermediate and require retrotranscriptase activity, and class II elements, or transposons, which move without the need for retrotranscription. In humans, class I TEs are grouped by length into SINEs and LINEs (short and long interspersed nuclear elements, respectively), alongside SVAs. SINEs include *Alu* and MIR elements; and LINEs comprise L1 and L2 sequences. Currently, there are records of class I elements being active on the human genome (*Alu*, SVA, and L1), but not of class II ones[1]. Class II elements are inactive in mammals because mutations have rendered their transposase gene non-functional[3].

Due to their mobility, TEs have relevant biological implications, and their detection is of great relevance for genetic and clinical studies. TEs are involved in the evolution of genomes[4,5]. SINEs influence gene regulation through interactions with regulatory pathways and mechanisms[6–9] and by altering methylation in surrounding regions[10].

Retrotransposition of active class I elements can affect gene function with pathological consequences[11]. This is the case for some hereditary breast and ovarian cancers, in which the responsible insertion can go undetected under usual protocols for molecular diagnosis[12]. This highlights the need for new diagnostic technologies. Inactive elements can also result in mutations, for example, by promoting DNA recombination between two elements with similar sequences[13].

The abundance and repetitive nature of TEs increases the difficulty of their detection in the human genome, particularly by using short read sequencing (SRS) methods, although some projects have used it[14]. Long read sequencing (LRS) allows the detection of TE insertions and reporting of the inserted sequence for further

[1]Departamento de Informática y Sistemas, IMIB-Pascual Parrilla, CEIR Campus Mare Nostrum, Universidad de Murcia, 30100 Murcia, Spain. [2]Servicio de Hematología, CIBERER-ISCIII, IMIB-Pascual Parrilla, Centro Regional de Hemodonación, Hospital Universitario Morales Meseguer, Universidad de Murcia, 30003 Murcia, Spain. ✉email: jfernand@um.es

study. LRS is more sensitive than SRS for the study of TEs, although both report coordinates with single-base precision[15,16].

There are multiple tools to detect TE insertions with NGS data[17]. Most of them process mapped reads, but some ones have been designed for unmapped reads, the latter requiring less computation[18]. MELT appears to be the most widely used tool[14,15] for the analysis of TEs using short reads. LRS-based variant callers can capture the complete inserted sequence[19] or provide sufficient information for subsequent reassembly[15,20]. Nanopore reads currently have relatively low accuracy[21], although consensus sequence generation tools[22] can mitigate this limitation. Similar error correction approaches have proven effective in RNA sequencing[23] and genome assembly, which enables SV analysis including TE insertions[24]. Analyzing the complete sequence of TE insertions facilitates the identification of elements in emerging subfamilies, such as SVA F_1[25], and provides crucial insights into their origins and evolutionary history[26]. These capabilities further demonstrate the value of LRS technology for comprehensive TE research.

Currently, researchers face a scarcity of LRS-compatible bioinformatics tools for TE analysis. For non-human organisms, available options include the TrEMOLO pipeline for *Drosophila melanogaster*[27] and VariantDetective for bacterial variant detection[28]. In the human context, xTea[15] detects TE insertions with both long and short reads. GraffiTE[29] employs graph genomes but lacks gene or enrichment annotation functionality.

In this work, we introduce RetroInspector, a comprehensive pipeline for detecting, characterizing, and annotating TE insertions and deletions using nanopore sequencing data, advancing the current state of the field. RetroInspector enhances our previously published analysis framework[20] through streamlined implementation, improved usability, and enhanced reproducibility. We also present findings from multiple datasets analyzed with our pipeline, demonstrating its practical applications and effectiveness. PALMER[30] and GraffiTE[29] served as comparative benchmarks for our pipeline.

Results

We have applied the RetroInspector pipeline to four distinct datasets comprising a total of 40 samples. The first dataset consisted of 24 samples sequenced by PromethION; the second one, of three samples (HG00733, HG00514, and NA19240) from HGSV (Human Genome Structural Variation Consortium)[31,32]; the third one, of two samples from monozygotic twins, also sequenced by PromethION. The fourth dataset consisted of a subset of 11 cases used for the HGSVC2 study[24].

The first dataset was used to design the pipeline and ensure that output by any step would be properly formatted to function as input for the next. The second dataset was used to assess the performance of multiple variant callers. The third one was used to test the accuracy of the pipeline on two samples with a high degree of similarity. The fourth dataset was used to benchmark the genotype process.

In this section, we present the metrics and the summary of the results obtained with RetroInspector, which are also compared with the results obtained with state-of-the-art tools. In our experiments, we have tested three combinations of variant callers, and three different thresholds for supporting read evidence. Additionally, we have compared the performance of two criteria for filtering variants, namely, lax and strict. As a last factor, we show how the margin of error used in the comparison against the benchmark's truth set (100 bp) is more stringent than the value used in other works (500 bp). The performance metrics were calculated for the variants obtained by applying the strict criterion, using cuteSV and Sniffles2 with at least 3 supporting reads, which are the default (and recommended) settings of RetroInspector, unless otherwise specified. These settings were used by default in all the experiments that we report after the benchmark results. The detailed list of settings can be consulted at the pipeline's repository, which includes a template configuration file with default settings.

Validation and benchmarks

The performance for variant calling had been previously evaluated by PCR amplification and sequencing on the results from our previous workflow, of which RetroInspector is the streamlined version. We have proved the reliability of the results generated by this pipeline by validating several of the reported insertions. We have confirmed the results generated with our previous workflow with specific PCR amplification and Sanger's sequencing in 14 out of 17 insertions selected for validation (82.4%)[20]. RetroInspector uses a different combination of variant callers than our previous work and called 13 out of these validated. Additional validation was performed with four of the variants detected by RetroInspector but not with the previous iteration. Two of such insertions were experimentally confirmed. The primers for a third one rendered amplification in both case and control. The last one did not yield a PCR product. In summary, 75% of insertions identified by our pipeline that were tested experimentally were validated (15 out of 20, since one of the previous 17 is not part of the current results).

To determine the best combination of variant callers, we used a truth set with three samples with SVs comprehensively characterized using multiple techniques[31], with a total of 5,630 reported TE insertions on these samples. This dataset also lists which insertions are related to TE elements. The following section details the concordance with the dataset and the slight difference in classification of insertions as TE insertions.

The programs tested were cuteSV, SVIM and Sniffles2. Although the three combinations performed similarly, cuteSV and Sniffles2's results were in better agreement with the HGSV dataset. The lax criterion found the most insertions from the set, but also found more insertions that were not in the benchmark set; although the loss in precision was lower than the gain in recall (Table 1).

Precision was above 80% for the subset classified as TEs by the HGSVC study, and above 85% when considering all insertions. When examining retrotransposition motifs, precision increased to 85.51%, as detailed later. Recall exceeded 85% for TE insertions, while remaining below 20% for the complete insertion set, because most insertions are not TE-related (17.72% of the insertions in the truth set are TE insertions) (Table 1 and Table S1).

Combination	Criterion	Precision	TE recall (INS recall)	F_1
cuteSV & sniffles2	Strict	80.90% (85.63%)	87.10% (16.34%)	83.89%
cuteSV & sniffles2	Lax	70.34% (75.58%)	90.36% (17.20%)	79.10%
cuteSV & SVIM	Strict	78.39% (83.22%)	87.37% (16.43%)	82.64%
cuteSV & SVIM	Lax	65.46% (70.75%)	91.92% (17.60%)	76.47%
SVIM & sniffles2	Strict	79.27% (84.81%)	89.18% (16.91%)	83.94%
SVIM & sniffles2	Lax	67.51% (73.04%)	91.99% (17.63%)	77.87%
cuteSV & sniffles2 + hallmarks	–	85.51%	71.81%	78.07%
cuteSV & sniffles2 + RE = 5	Strict	82.25% (87.03%)	86.41% (16.20%)	84.28%
cuteSV & sniffles2 + RE = 5 + hallmarks	–	87.06%	71.35%	78.43%
PALMER	–	71.77%	52.61%	60.72%
GraffiTE	–	32.33%	96.18%	48.40%

Table 1. Benchmark results comparing combinations of variant callers and criteria. The rows for "strict" and "lax" specify a different callset produced according to that criterion. (The strict dataset contains only variants that passed the strict criterion on at least one sample.) Precision: percentage of TE insertions found present in both the pipeline's and HGSVC's results among the number of the pipeline's calls. Recall: percentage of TE insertions called by the pipeline among those present in the HGSVC dataset. For each cell, the first number is obtained by selecting only the insertions labeled as "ME" in the HGSVC set, while the second one includes all HGSVC's insertions. F_1-score is shown for the TE subset of the gold standard. RE: read evidence threshold. Additionally, metrics are shown for the addition of retrotransposition hallmarks' presence, the use of a higher read evidence threshold, and for other TE software: PALMER and GraffiTE.

Callers	Precision	Recall	F_1-score
cuteSV & Sniffles2	96.99%	96.65%	96.82%
cuteSV & SVIM	95.37%	93.68%	94.52%
SVIM & Sniffles2	95.52%	93.81%	94.66%

Table 2. Benchmark results for genotyping.

Increasing supporting read evidence thresholds enhanced precision. For instance, setting the threshold at five supporting reads yielded a precision of 82.25% (87.03% when considering all insertions) for cuteSV and Sniffles2. While recall decreased to 86.41%, the F_1-score marginally improved to 84.28% compared to the previous threshold. With the minimum of five supporting reads, the combination of Sniffles2 and SVIM was still marginally better, with an F_1-score of 84.46% (Table 1 and Table S2). Raising the threshold to seven supporting reads resulted in only a minimal precision increase (to 82.67% or 87.64% for all insertions, using cuteSV and Sniffles2), accompanied by further reductions in recall (83.98%) and F_1-score (83.32%) (Table S2).

Filtering by retrotransposition features further improved precision for these higher thresholds, to 90.38% and 90.94% for 5 and 7 supporting reads, respectively. However, this approach penalized recall (71.35% and 69.25%, for 5 and 7 reads, respectively) and the F_1-score (78.43% and 77.37%, for 5 and 7 reads, respectively) (Table 1 and Table S2).

When considering a broader distance margin of 500 bp for consensus comparison (matching Truvari's default setting), the results improved. For instance, the combination of cuteSV and Sniffles2 reached a precision of 82.79%, recall of 88.72%, and a F_1-score of 85.66% (Table S3). Applying these more standard parameters, filtering by retrotransposition characteristics further increased precision to 91.17%. Similarly, higher read evidence thresholds also yielded better results (Table S4).

Genotype benchmark

RetroInspector calculates genotypes from read evidence and coverage of the region supporting the insertion. These genotypes are then used to filter the variants that are found to have a 0/0 genotype (homozygous for the reference allele). To test the accuracy of this process, a subset of the HGSV2 cohort[24] was used. RetroInspector's results and the truth set were highly concordant, with precision exceeding 95% for all combinations of variant callers, and the best results were 96.99% when using cuteSV and Sniffles2 (Table 2).

Edge false negatives of calling and merging

To further validate the pipeline, we analyzed samples from two monozygotic twins using the comparison features of RetroInspector. Initially, with a 60 bp merge limit, 94 variants (3.87%) were found exclusively in a single sample. Examining the insertion list, we identified pairs of insertions unique to each sample, characterized by similar lengths and identical types. Expanding the maximum merging distance to 80 bp reduced the sample-specific insertions to 82 (3.36% of total).

We manually examined the alignments near these variants using IGV to investigate the source of discrepancies, which could result from merging errors, insufficient read coverage, or genuine *de novo* mutations. Our analysis revealed that in most cases (44 variants, 53.66%), the non-shared variants were located in noisy genomic regions characterized by multiple inserted sequences of varying sizes within the reads, coexisting with alleles without insertions, which precluded the possibility of distinct insertions at the same site. These variants also exhibited dispersed coordinates. A smaller subset of differences (18 variants, 21.95%) stemmed from underrepresented inserted alleles, arising from two distinct scenarios. In the first scenario, the depth coverage was extremely low in one sample for the region containing the insertion, preventing the minimum number of supporting reads required from being detected. In the second scenario, while regional depth coverage was adequate, the inserted samples remained underrepresented. In both scenarios, the number of supporting reads did not reach the threshold of 3 reads. A minority of ten variants (12.20%) could not be matched due to low sequence similarity between insertions. The final ten variants (12.20%) were not identified by either program in one sample, despite apparently sufficient supporting reads.

Subsequently, we investigated whether the variants with few reads were reported by Sniffles2, which employs a read fraction-based approach instead of a fixed threshold, by examining the output files before our processing. It reported one out of the 18 in this category as homozygous even in the sample with minimal coverage.

Time usage and scalability

To evaluate the impact of specific analyses on runtime, we measured the duration of distinct phases within the pipeline across runs comprising our 24-sample and the 3-sample HGSVC cohorts.

Alignment was the most time-consuming stage, with a mean duration of 231.88 ± 32 min per sample. Sorting and indexing the alignment file required 12.78 ± 1.31 min per sample. Running Sniffles2 took 7.29 ± 0.33 min per sample, with an additional 2.10 ± 0.15 min per sample for sequence reassembly. CuteSV required 28.01 ± 0.90 min per sample, and sequence reassembly for its reported insertions took 8.2 ± 0.29 min per sample. Genotyping insertions and deletions consumed 37.83 ± 3.59 min per sample, with 98.3% of this time dedicated to calculating genome-wide coverage.

The remaining steps were conducted across samples rather than per-sample. Comparing runs with 3 and 24 patients revealed that insertions merging took 1.76 min for three patients and 15.71 min for 24 patients. RepeatMasker runtime was 47.84 and 84 min, respectively. R analysis took 10.13 and 5.87 min for the two cohort sizes.

We also benchmarked PALMER, another long-read TE detection software, to use as a reference point, which analyzed the three HGSVC cases in a mean of 1,535.05 ± 1,175.95 min, totaling 4,605.15 min. In comparison, RetroInspector processed the same samples in 1,005.14 min. It is crucial to note that we utilized the BAM files generated by RetroInspector; if these were not pre-existing, PALMER's total processing time would increase.

Identification of retrotransposition hallmarks

To explore additional filtering criteria that could enhance specificity, we investigated two key retrotransposition features: polyA tails at the inserted sequence's terminus, and the presence of the consensus motif for L1's endonuclease target sequence. We also run PALMER and GraffiTE, which search for similar features[29,30], on the three HGSV samples for comparative analysis.

In our tests, PALMER demonstrated lower performance, achieving a maximum F_1-score of 60.72%, with a precision of 71.77% and recall of 52.61% (Table 1). Using the recommended settings from the GitHub documentation yielded the best results (Table 1); while more strict filters, requiring more reads to preserve a variant call, progressively reduced recall and increased precision (Table S5). GraffiTE showed high recall (96.18%) and low precision (32.33%), achieving a F_1-score of 48.40% (Table 1).

Our search for retrotransposition hallmarks improved precision at the cost of recall. The target sequence was detected in 84.46% of true positives and 57.99% of false positives. While the proportions differed significantly (χ^2, $p < 2e{-}16$), the discrimination was imperfect. Considering that the identification of the consensus sequence allows for divergence, we examined whether different ratios of similarity would result in a better threshold. Using a ROC curve, the area under the curve (AUC) was low at 0.6774, offering minimal discriminatory value (Fig. S1A).

Similarly, polyA tail presence proved an unreliable indicator. A tail of at least 4 bp (with gaps not contributing to tail length) was detected in 92.67% of true positives and in 77.50% of the false positives, representing a significant difference (χ^2, $p < 2e{-}16$). Additionally, other values of the tail's length did not perform better (AUC=0.561), and neither did by removing the gap penalty (i.e., counting non-A bases for the length of the polyA tail, AUC=0.5106) (Fig. S1B and S1C). While these hallmarks did not enhance the pipeline's overall performance, they marginally increased precision (Table 1).

Identification of TE insertions and deletions

Tables 3 and 4 present the results obtained for the 24 case cohort, which consist of patients with antithrombin (AT) deficiency. AT deficiency is a rare, autosomal disease most commonly caused by mutations in *SERPINC1*[33]. A total of 6,714 TE insertions were identified,amounting to 48,963 occurrences across patients, of which 43,288 met the strict criterion. Considerably fewer class II insertions were found, 21. The most populated categories of class I insertions were Alu and SVA insertions (Table 3), TEs that are active in humans. RetroInspector reported an intronic SVA insertion in *SERPINC1* in one of the patients. However, it did not do so for another patient, as their data only contained a single supporting read.

Among the SVA, the most common insertions were classified as SVA F (598 of 1,189 total SVA insertions), with 6 additional SVA F_1. The rarest SVA group was SVA C (Table 4). The newer family of SVA elements, SVA F_1, is not included in Dfam, and is in turn reported by RepeatMasker as another type of SVA, normally as SVA

Repeat family	Repeat subfamily	Unique	Total (lax)	Total (strict)
DNA	hAT	10	141	135
	TcMar	11	126	105
	CR1	1	24	24
LINE	L1	467	4197	3,781
	L2	6	71	64
LTR	ERV1	256	1901	1,435
	ERVK	30	313	286
	ERVL	68	593	449
Retroposon	SVA	1189	8180	4,741
SINE	Alu	4,675	33,395	32,246
	MIR	1	22	22
Total	–	6,714	48,963	43,288

Table 3. Insertion count by repeat family and subfamily on the 24 sample cohort.

Name	Unique	Total (lax)	Total (strict)
SVA A	172	1,111	679
SVA B	64	574	402
SVA C	49	366	271
SVA D	96	668	384
SVA E	204	1,458	852
SVA F	598	3,971	2,128
SVA F_1	6	32	25
Total	1,189	8,180	4,741

Table 4. SVA count on the 24 sample cohort.

F. RetroInspector examines SVA insertions for the *MAST2*'s thumbprint[25] and, if found, reports them as SVA F_1 (Table 4).

Figure 1 shows RetroInspector's graphical output. First, it plots the count of TE insertions and their gene annotations (Fig. 1A), allowing users to visualize the genome-wide distribution, and check, for example, if the sex chromosomes carry more TE insertions, as is expected for L1 elements[34]. Second, a Manhattan plot of AFs is displayed (Fig. 1B), which can help to determine whether a cohort contains polymorphic insertions with high AFs. These two sets of plots are repeated for TE deletions. Third, enrichment results are presented as connection between categories for each GO group (Fig. 1C). Fourth, enrichment results are shown as bar plots (Fig. 1D) for each GO group and the other ontologies, which involve cancer (NCG) and other diseases (DO).

Reassembly of the inserted sequences

Given that not all variant callers report a consensus for insertions, our tool retrieves reads that support the insertion, and reassembles the affected region in order to obtain a new sequence.

The reassembly process excludes reads with a considerably shorter insertion (length difference greater than 15% of the reported insertion length) to minimize the influence of noise, although callers do consider these reads to support the evidence (Fig. 2). These reconstructed sequences are also employed by RetroInspector's sequence-aware variant merging.

The final report is divided into two sections, one for class I and class II elements. Each contains gene annotation (with affected gene and type of affected sequence, such as coding sequences or promoters), as well as the distribution across chromosomes. It also includes a comparison against the Indigen[14] dataset[35], which can be used to determine whether a TE insertion has been previously described.

Genotyping TE insertions and deletions

RetroInspector uses coverage and read support data to compute genotypes for insertions. These genotypes can then be used to determine allele frequencies within the sample, which are shown in a Manhattan plot (Fig. 1). The genotype is also compared to the strict and lax datasets, which allow users to evaluate both sets. Additionally, it is used to filter out insertions that are genotyped as homozygous for the reference allele in all cases, that is, 0 inserted alleles, or no insertions in neither of the homologous chromosomes, which are most likely false positives in regions with noisy alignments and high coverage[20]. This occurs in regions where depth coverage reaches values in the hundreds, in which read errors can reach an otherwise reasonable threshold for supporting reads. This is not the case for the variants' genotypes, which are calculated using the coverage of their region.

Using the genotype as an additional filter is useful to remove likely erroneous calls. On the 24 sample cohort, the stringent criterion left 902 TE insertions (12.9%) which were genotyped as no inserted alleles in at least one

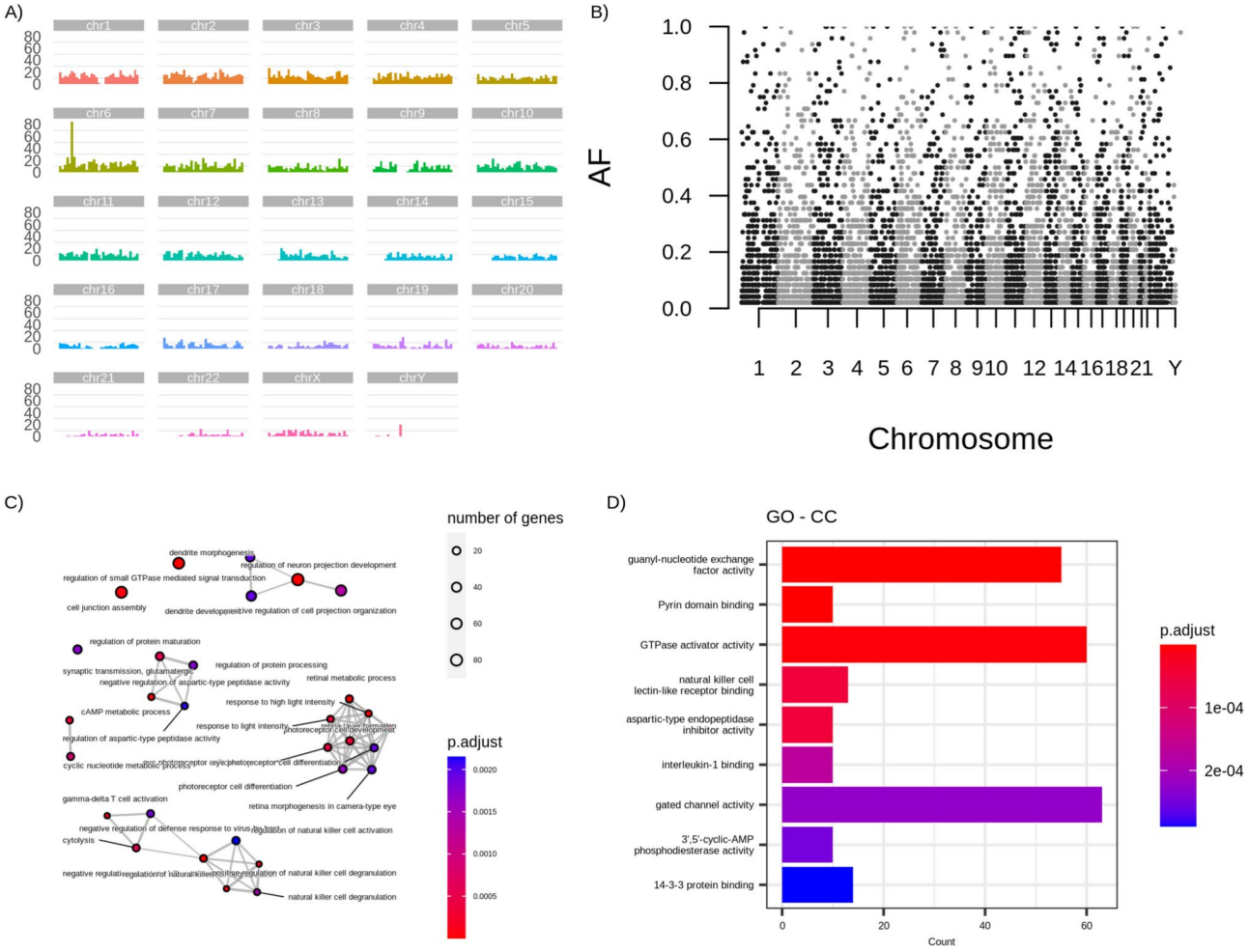

Fig. 1. Examples of plots generated by RetroInspector with the 24 patient cohort. (**A**) Count of TE insertions (merged across patients) on each chromosome. Similar plots are produced for class I and class II insertions separately, and for TE deletions. (**B**) Manhattan plot for TE insertions, with AF calculated within the sample. A similar plot is produced for deletions. (**C**) Plot showing enriched GO categories for molecular function, and their connection by genes in common. This plot is also produced for the other GO categories. (**D**) Barplot of enriched GO "biological process" categories. Color indicates p-value and bar size indicates number of genes. Similar plots are produced for other GO categories and for the other ontologies, if significant results are found.

sample, adding up to 1423 occurrences across samples. Regarding these, it is relevant that 268 insertions were genotyped as no inserted alleles on all samples they had been detected in. This number was higher for the lax criterion (881 unique insertions). RetroInspector discards these variants for both criteria, so that they are not considered in any other process.

The reference sequence for the human genome contains hundreds of thousands of TEs. RetroInspector matches their coordinates against the deletions reported during the variant calling process, and performs the same genotyping described for insertions, also removing erroneous calls and calculating AFs.

Comparison of samples

RetroInspector can also compare pairs of samples. It reports TE insertions found in common and not in common between the samples, as well as their annotation.

As an example, we have compared two of the samples from HGSV, NA19240 and HG00514. RetroInspector found 3,310 unique TE insertions across both samples, of which 870 were present in both.

Regarding sample comparison, two unrelated samples of different ancestries (NA19240's is Yoruban, HG00514's is Han Chinese) share 870 TE insertions, while they carry 3,310 insertions not in common (Table 5). For active TEs (*Alu*, SVA, L1), the fraction of non-shared insertions is higher than for inactive ones (75% against 50%, Table 5).

This feature has interesting potential applications such as comparing the TE profile of changes in individuals over time, cases and controls, etc. Comparing the functional enrichment of genes in two samples can also be useful, as TEs can play a role in multifactorial disease. We observed that different functions were enriched. In HG00514, activities related to molecular signaling (cAMP hydrolysis and binding to phosphatidylinositol

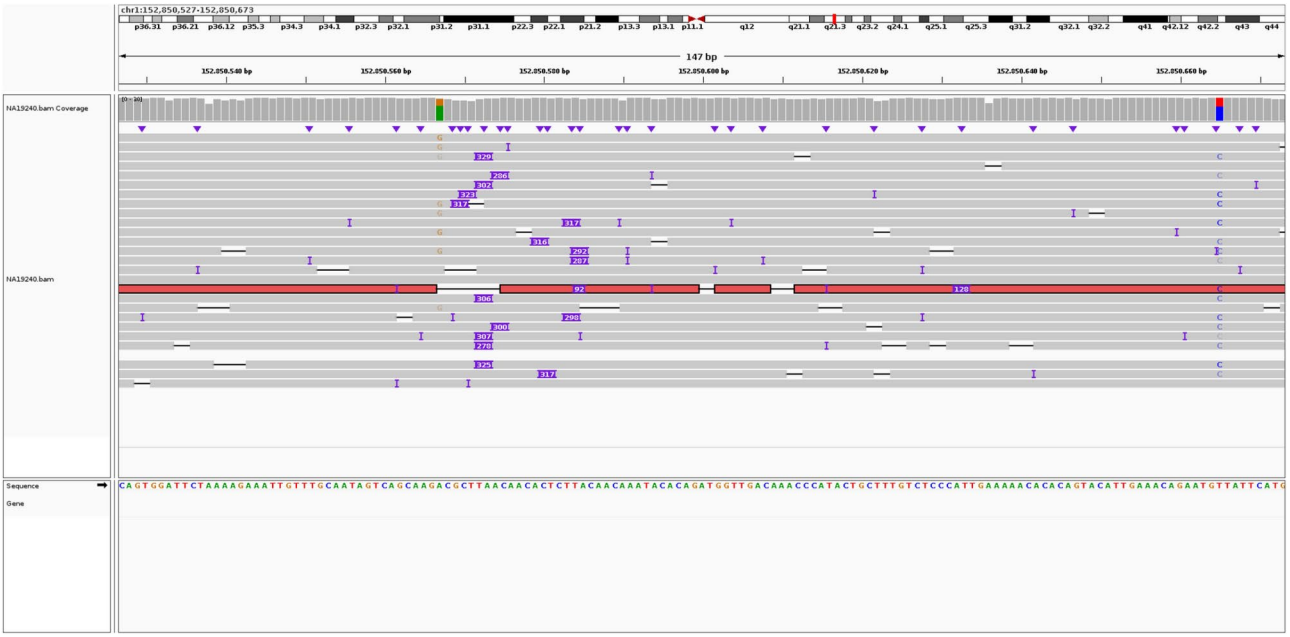

Fig. 2. View of the alignments around an *Alu* insertion found in NA19240's genome. The insertion is reported to be 301 bp long by cuteSV, although one of the supporting reads (marked in red, read ID "e7811730-08a4-4eaa-8827-bb981538051f") only contains a 92 bp and a 128 insertion. RetroInspector discards inserted sequences with a length difference greater than 15% for the allele reconstruction step.

Class	Family	Subfamily	Number of unshared insertions	Number of shared insertions
Class I	DNA	hAT	3	2
		TcMar	2	2
Class II	LINE	L1	253	152
		L2	3	2
		CR1	1	0
	LTR	ERV1	60	48
		ERVK	12	8
		ERVL	16	18
	Retroposon	SVA	182	87
	SINE	Alu	1,827	630
		MIR	2	0
Total	–	–	2,361	949

Table 5. Insertions in common ("shared") and not in common ("unshared") between samples NA19240 and HG00514.

phosphate and calmodulin) were enriched (Fig. 3A), whereas the results of the enrichment analysis of NA19240's own insertions affect genes related to the class II MHC (major histocompatibility complex) (Fig. 3B).

Discussion

TE insertions are a significant source of genomic variability. Researchers are increasingly interested in analyzing these variants at the genomic level and exploring their potential phenotypical implications[36,37]. Long-read sequencing (LRS) has emerged as the most effective method for detecting TEs in the human genome[15,16], yet a comprehensive tool for TE analysis and annotation has been lacking.

RetroInspector addresses this gap by providing a streamlined process for discovering and studying TE insertions. Built on Snakemake, the tool offers automatic dependency installation, ensuring both reproducibility and user-friendliness. The pipeline leverages multi-threaded computing to process data in parallel, utilizing existing tools like minimap2, cuteSV, and Sniffles2, as well as custom-developed scripts, the latter explained in the "Methods" section. For single-threaded programs, Snakemake optimizes resource allocation by running instances in parallel when sufficient computational resources are available.

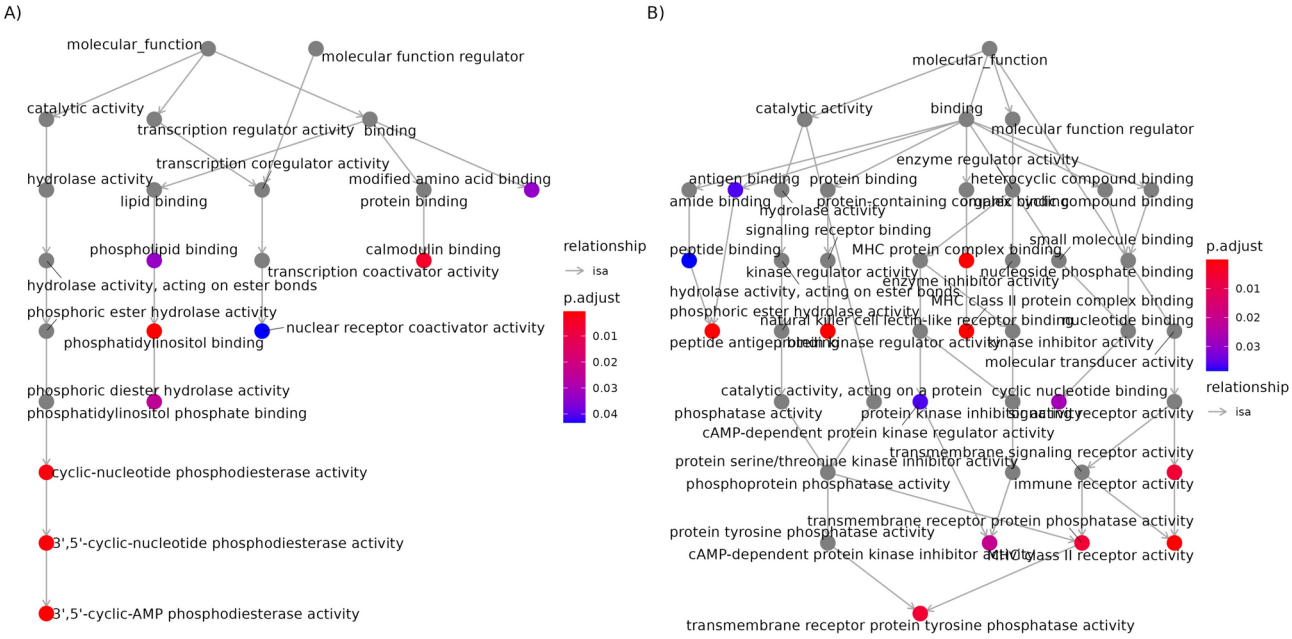

Fig. 3. GO hierarchy of enriched terms (only significant functions are colored). (**A**) Enrichment for intragenic insertions present in HG00514 and not in NA19240, and (**B**) enrichment for intragenic insertions present in NA19240 and not in HG00514.

The results of our pipeline include inactive elements of both class I and II, presenting a more difficult interpretation than insertions for the ubiquitous and highly active *Alus*, which have an insertion frequency every 20 human births[38]. Critically, we have validated insertions from these inactive elements to confirm that they are not the result of alignment errors from malformed reads[20]. They are likely not the result of (retro)transposition, either, as they are incomplete elements, likely arising from ectopic recombination, or representing polymorphic ancestral copies that are not part of the reference genome.

Generation of biological annotation

Transposable element (TE) sequences impact human health through various mechanisms, including direct insertional events[11,33] and by facilitating genetic recombination[3]. Moreover, both *Alu* and SVA elements significantly influence transcription and methylation patterns, with *Alu* expression specifically linked to cancer[10] and aging processes[39]. Consequently, investigating TE variations across individuals is crucial for developing a comprehensive understanding of the human genome.

The annotation and enrichment analysis can inform users of genes and functions affected by TE insertions. Both *Alu* and SVA elements affect the genome by interfering with RNA and creating differentially methylated regions[8–10], which could be of interest for cancer diagnosis[40].

Using nanopore reads, RetroInspector offers a comprehensive workflow that enables researchers to create alignment and variant files, extract transposable element (TE) insertions, assemble inserted sequences, perform annotation, and conduct enrichment studies. The tool is also valuable for comparing two samples, such as control-proband pairs, or related individuals like twins, by identifying shared and unique TE insertions. The pipeline generates a user-friendly report, standard variant files (VCF), and intermediate R files that provide researchers with a flexible foundation for further analysis. A key innovative feature is its approach to merging variant sets from different callers or samples, which takes inserted sequences into account. This is critically important because retrotransposition tends to target specific genomic hotspots[41], and insertions at similar coordinates across patients may actually represent distinct genetic events. By distinguishing between seemingly similar insertions based on their specific sequences, RetroInspector enables more precise characterization of population-level TE variations and more accurate allele frequency calculations.

Measurement of performance

Other reproducible tools have been developed to detect TE insertions in other species' genomes[27]. With RetroInspector, we present an easy to use and well-performing tool that allows for TE insertions and deletions, as well as their characterization, on humans. We have tested RetroInspector for both accuracy and time consumption. The calling of insertions has a precision in the range of 80–90%. Using higher thresholds for read support results in an increase of precision, as so does searching for retrotransposition features, although both negatively affect recall.

While all current variant callers perform well for the purpose of TE detection, we have chosen Sniffles2 over SVIM. The difference in performance is minimal, as shown in "Results" (Table 1), but Sniffles2 can be run multi-threaded, while SVIM cannot at the moment, which is relevant for large datasets.

When compared to PALMER, it performed considerably better (difference of F_1-score of 20%, ∼10% in precision and ∼30% in recall), although it is important to keep in mind that PALMER is designed for complete TE insertions, which explains the lower recall, but not necessarily the low precision, which could stem from an incomplete truth set. A similar argument could be made for GraffiTE's results, which obtained a very high recall, but low precision.

The consideration of retrotransposition hallmarks improved the precision of the results, with precision values above 90% for higher supporting read evidence thresholds, with a negative impact on recall, resulting in an overall lower F1-score. This may be explained by several factors, for example, the consensus motif for endonuclease cleavage is short and does not seem to be the only factor determining targeted integration[42], and insertion sites can be dissimilar to the motif[43]. For polyA tails, its low value as a classifier can be explained by two factors. First, the publicly available HGSVC data is a few years old, meaning it was basecalled with an older basecaller, and even after reassembly, sequences may be imperfect, particularly for homopolymers[16]. Second, after an insertion event, through subsequent generations, the inserted sequence may be altered.

For time metrics, the most relevant datapoint is that alignment, which is the starting point of most workflows and takes the most time. The length of this stage will depend on the number of samples and depth coverage, and not on the implementation of RetroInspector. In our tests, RetroInspector took less time than PALMER, while generating a more complete analysis. The long PALMER runtimes do not seem to be due to limitations in our infrastructure, since other studies also mention that it requires more than 48 hours per sample[44], which is longer than it required in our case. Regarding only RetroInspector, we compared two runs with differently sized datasets, one having eight times the size of the other one. We found that among the multisample steps, the ones that could scale badly with the amount of samples, the RepeatMasker step, the longest one only doubled in time, probably because we run RepeatMasker after merging insertions across samples, so it does not have to analyze duplicates. The merging of samples did have an eight-fold increase in time, which, since it coincides with the increase in sample number, points to a linear scale, and the addition of ∼1 min per sample is negligible compared to the cost incurred by other parts. The R analysis was shorter with the larger dataset, perhaps due to a different number of triggers of garbage collection, connection to online resources, or other processes with a similarly unpredictable runtime.

Consideration and interpretation of biological aspects

The analysis of twins invites to several points of discussion. First, it reveals that the pipeline is accurate and the differences in identical samples are caused not by tools' choice or parameters, but by random sampling of reads during sequencing, and this differences are minimal (<4%). This is similar for allele dropout, which is a problem in genetic diagnosis[45]. Since nanopore sequencing does not require the preparation of a DNA library, allele dropout during library preparation is not a problem, and the cases of uneven sequencing of alleles and regions with anomalously low depth coverage seems to be very low (<0.1%). Second, the noise in some regions is probably produced by errors in the basecalling due to repetitive sequences, and does not correspond to mutations. Third, using allele fractions instead of read support, following Sniffles2's method, to filter variants, a single call was rescued from a low-coverage zone. However, most did not, and the results produced by our pipeline, which filters mostly by supporting reads, are still accurate. RetroInspector indirectly takes allele fractions into account, which are based on the number of reads that support a variant, since insertions genotyped as homozygous for the reference allele on all samples are discarded. Finally, it allows to explore which margin errors for merging intra- and inter-sample callsets yield better results, and what errors are currently unavoidable.

Retroinspector has been designed for long read sequencing, particularly nanopore sequencing, and its characteristics have been considered at the design of every stage. For example, the variable reads are individually analyzed to filter out sequences of greatly differing length that callers report as supporting reads. While some variant callers can return consensus sequences, removing sequences dissimilar to the rest of the reads can result in a better consensus. Additionally, merging insertions considers the sequence of the insertions, while other programs, like SURVIVOR or GraffiTE (which uses SURVIVOR) do not. The reassembled sequences and insertion coordinates can also be used to annotate characteristics of retrotransposition, which increases precision to values above 90%. We also run an additional TE detection step for SVA F_1, which could be expanded to other elements if the need arises. Although retrotransposition features did improve performance as measured by F_1-score, exploring target site duplications could be a future improvement, whose implementation would be facilitated by less variance in coordinates which still exists, as shown here by the twin analysis.

After variant detection, RetroInspector conducts additional analyses, such as gene enrichment and calculation of allele frequencies within the cohort. Results are presented into standard formats, such as VCF files, and also a human-readable report. The VCF files contain enough information so that they can be combined to files from other cohorts and AFs can be estimated more accurately.

Some of these steps, however, would make RetroInspector overreliant on a particular reference version, since when new versions of the reference genome become available, they get adopted progressively, so gene annotation may not be available when the sequence is released. Using newer references can be useful, since they may allow for alignment against previously erroneous sequences that contain genes of clinical interest[46]. To solve this, RetroInspector allows a shorter analysis that omits annotation, but keeps the rest, and in exchange allows for alternative reference versions. The resulting files are standard formats, meaning that they can be easily used in further downstream analysis with implemented support for other references.

RetroInspector seems like a promising tool to uncover the variation on human genomes. If we look at projects that study TEs using NGS tools[47], we can compare the amount of insertions reported. For example, the Indigen project sequenced 1,021 genomes and discovered 9,239 *Alu* insertions[14]. Another study, using their software with 90 samples, reports 9,342 TE insertions, including 64,383 *Alu* elements[48]. On 24 samples, our tool reports, for example, 4,724 *Alu* insertions (of which 3,739 are 280 bp, with the length of a full element, which may have

been the target for the other studies). With a small fraction of the samples, the amount of TEs is comparable to the other studies. Additionally, Yu et al. report 5 insertions in protein coding sequence; similarly, we have found 7. This points that nanopore sequencing data, as analyzed by RetroInspector, can uncover a hidden trove of genetic variation, and that these variants may be of clinical importance.

In conclusion, RetroInspector is a complete, tested and user-friendly pipeline capable of completely examining human TE variation on a sample, from their detection to their biological implications. It has the potential to help researchers to better understand a particularly complex aspect of human genetic variation. RetroInspector supposes an improvement in the field of TE discovery and analysis with long-read sequencing data, since RetroInspector outperformed PALMER and GraffiTE in our experiments. A limitation of our comparative evaluation is that we were not able to use xTea, whose performance according to the literature is promising. Additionally, it has been tuned to work with nanopore data to maximize precision and recall, and has additional functionality such as allele frequency calculation and gene annotation. As a Snakemake pipeline with its dependencies defined in conda environments is simple to deploy, and it can be used in even more environments with its docker container.

With the biological importance of TEs, which were first thought to be inert or "junk" DNA, becoming more apparent, they may become target for GWAS or other massive functional studies that will tie connections between variants and effect, which require a systematic and reliable method for TE detection and characterization.

Methods

RetroInspector provides a pipeline for the identification and characterization of TE insertions and deletions using long-read sequencing as input data.

It integrates the workflow of a previous TE study[20] as its foundation. It has been improved with new features and uses different software and libraries (such as Sniffles and spoa) for several steps. It features a more robust handling of input and connection between steps, and is now implemented as a Snakemake pipeline[49].

Figure 4 shows the steps included in this pipeline, which can be grouped in the following major tasks: alignment, variant calling, identification and annotation of TE insertions and deletions, generation of results. This pipeline can be applied to the data obtained by applying one sample, but it can be also adapted to perform a comparative analysis of pairs of samples. In this case, the pipeline is executed once per input sample, and the results are compared. Next, we describe the main tasks of the RetroInspector pipeline.

Alignment

Alignment is the process through which we search the region of the reference genome that provides the best match for each basecalled nanopore read, as a way for reconstructing the genome of the sample. In our case, the basecalled nanopore reads are aligned against the reference human genome version GRCh38 with minimap2[50], and the alignment is sorted and indexed with samtools[51]. The version of the human genome was selected by its annotation availability, although other versions are compatible with RetroInspector if gene annotation is omitted. This step generates the alignment in BAM format.

Variant calling

The variant calling process identifies the difference between the sample's genome and the reference genome. This process takes the alignment obtained in the previous step as input.

The variants are called with cuteSV[52] and Sniffles2[19], generating VCF files. This combination of programs has been selected based on existing benchmarks[53] as well as testing for recall of insertions (results detailed in the Validation section of "Results"). This step generates two VCF files, one per variant caller. Variant calls are first filtered by read evidence, which is one of RetroInspector's parameters. We have run RetroInspector with this value set for 3, 7, and 5 for the HGSVC cohort, and with 3 for the other datasets. SVIM[54] is also available as a variant caller and was evaluated as described in the benchmark section.

We filter these VCF files to edit malformed variant records that would otherwise obstruct subsequent steps, mainly variants with negatives or 0 coordinates, which sometimes are reported in decoy chromosomes and other scaffolds. RetroInspector analyzes the insertions reported by both callers to propose the final set of variants, which involves a double selection: first, selecting insertions that are TE related; second, selecting which insertions to keep as positives or to discard as false. RetroInspector provides two criteria for keeping positive results. The *stringent criterion* requires both callers to report an insertion on a sample in order to accept it (intersection of insertions) at least one of them surpassing the read evidence threshold set by the user. The *lax criterion* only requires one caller to report it (union of insertions), the read evidence threshold still applying. Both criteria are executed and their results are provided to the user.

Identification of TE insertions and deletions

This process takes as input the variants called in the previous step and finds which correspond to insertions and deletions, which require different processing. The pipeline branches off for insertions and deletions.

The left branch (Fig. 4) processes insertions. The inserted sequences are reassembled retrieving them from supporting reads using pysam[55], ensuring that the reads contain an insertion of similar size to the reported insertion (Fig. 2), and reassembled using the Python bindings for library spoa[22], which is designed to create consensus sequences from noisy reads. A difference of 15% in size is allowed to account for nanopore's error rate[16].

After that, insertion calls within the same sample are merged into one callset per sample, then they are genotyped. Genotyping is performed with a reimplementation of the Sniffles2's algorithm[19] using coverage data generated with mosdepth[56].

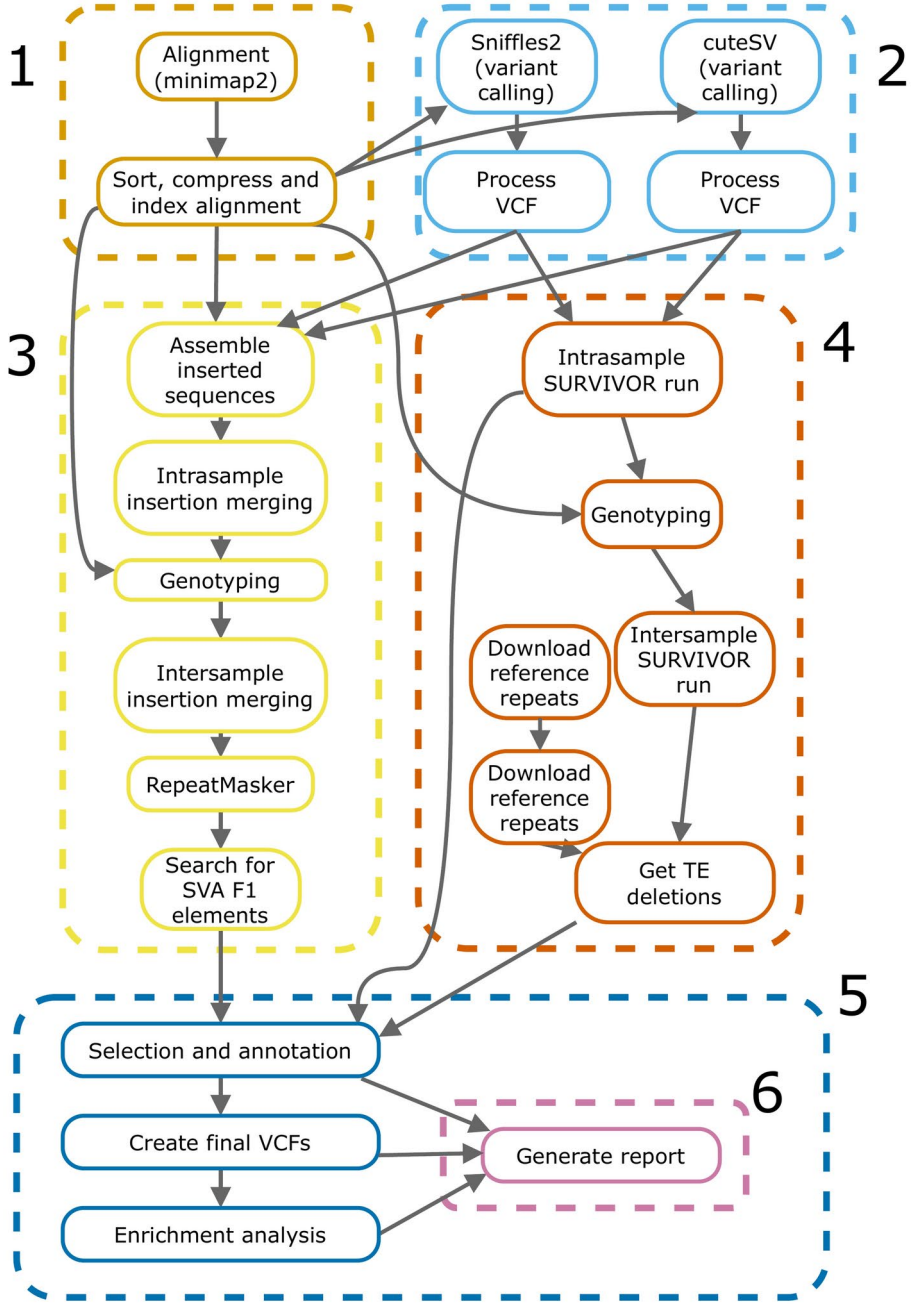

Fig. 4. The RetroInspector pipeline, consisting of 5 phases, numbered 1–5 and marked in the same color. (1) Reads are aligned using minimap2, and the alignment is sorted and indexed. (2) Variant calling is carried out with cuteSV and Sniffles2. Two processing paths diverge from here. (3) The left branch processes insertions. For this, inserted sequences are reconstructed. After that, insertion calls within the same sample are merged, then they are merged across samples. Next, TE sequences are identified by RepeatMasker, and an additional search for SVA F_1 elements is performed. (4) The right branch uses SURVIVOR to merge all variants and selects deletions that overlap TE sequences in the reference genome. (5) The results of both branches are used by the preparatory analysis to select variants related to TE and annotates TE insertions with gene information. The following step involves calculating coverage and genotyping TE insertions and deletions. Afterwards, an enrichment analysis is conducted on genes affected by TE insertions. (6) Finally, the report is generated. In light mode, which allows for alternative reference sequences, only steps 1, 2, 3, selection and VCF from 5 are performed.

Genotyping is also used to filter the results independently of other criteria. For this, variants genotyped as 0/0 are discarded. This can occur in noisy regions with exceedingly high coverage (hundredths or thousands of reads), in which the read number cannot function as a threshold.

The genotyping process was benchmarked with a 11-case cohort from data from HGSV2[24], which produced a curated set of genotyped variants. Only variants in autosomal chromosomes were selected, and to grade genotyping separately from variant calling, only variants present in the truth set were selected. To calculate performance metrics, genotypes concordant with the truth set were considered as true positives. For example, if the genotype from the truth set was 1/1, a genotype of 0/1 would count as one true positive, since it agreed on one allele, 1/1 as two true positives, since it agreed on both alleles, and 0/0, as no true positives. The number of predicted positives was determined to be the number of analyzed variants multiplied by two, to account for the fact that each variant has two alleles. The number of positives was the number of variants present in both sets, again multiplied by two.

After genotyping, insertions are merged across samples into one multi-sample callset. For merging insertions, we have implemented our own method, similar to SURVIVOR[57] in that it considers length and coordinates to compare insertions, although it also uses the same fields in the output to generate the same type of data. But, in addition, the sequence of the insertion (the only SV type processed in our method) is also compared with the Levenshtein distance, equivalent to a local pairwise alignment that allows for gaps and substitutions.

Insertions are determined to be TE insertions by searching the consensus sequence of the called insertions in the VCF files with RepeatMasker[58] on Dfam[2], a database of TE sequences. An insertion is accepted as a TE insertion when the RepeatMasker alignment covers 85% of its sequence. Some margin is given to account for the fact that insertions may also contain a duplicated insertion site and an extended polyA tail. An additional search for SVA F_1 elements is performed with BLAST[59] by comparing SVAs against the *MAST2* fragment that SVA F_1s contain[25].

The right branch (Fig. 4) processes deletions. It uses SURVIVOR[57] to merge all variants and selects deletions that overlap TE sequences by comparing deletions' coordinates to TE sequences present in the reference genome[60] and combining call sets performed with SURVIVOR. For example, if a sample carries a deletion spanning chr1:173,910,295–173,910,600, which overlaps the *Alu* present at chr1:173,910,298–173,910,599, that deletion would be selected at this step. The previous acceptance criteria regarding number of supporting reads and variant callers are also applied to these deletions.

The results of both branches are used by the R analysis to select TE variants and to annotate TE insertions. With the previously annotated genotypes, allele frequencies (AFs) within the studied sample can be calculated. The output for this steps consists of serialized data files.

Annotation of TE insertions

Gene annotation is the process through which we assign biological implications to variants. Through gene annotation, the insertion of interest are associated with information about whether they overlap any genes, and if so, if they affect an exon, intron, or a regulatory region. This task has been carried out in R[61], using annotatr[62].

Additionally, we perform a gene enrichment analysis of the variants of interest. Gene enrichment analysis consists of identifying the most relevant biological implications (e.g., molecular functions, diseases, biological processes, etc.) of the group of genes of interest. That enrichment is associated with statistical significance of the group of genes for the particular function, diseases, etc. We have implemented the Gene Enrichment analysis in R with the packages clusterProfiler[63] and DOSE[64], with data from Gene Ontology[65], Disease Ontology[66] and the Network of Cancer Genes database[67].

Other tools used include data.table[68], as a substitute for base R dataframes, qqman[69], for Manhattan plots, and surpyvor[70], as a wrapper for SURVIVOR.

Performance and time benchmarks

The presence of variants in the truth set was checked with SURVIVOR with a distance margin of 100 bp. Other SV studies use Truvari, which by defaults allow a distance of 500 bp, which results in better performance. We also run an additional comparison with a 500 bp limit, as shown in "Results".

In addition to RetroInspector, we also executed PALMER and GraffiTE, as detailed in the next section.

For measuring time, Retroinspector was run with the `--stats` flag. We compared the results of two different cohorts: the 3 samples from HGSVC and the 24 patient cohort on 31 threads. PALMER was run with GNU parallel, as its documentation recommends, which can also report time consumption. PALMER detects one of four TE types at a time, so it has to be run 4 times per sample. The time per sample was calculated by adding the longest time for each TE type on that sample. It was run with parallel suing 20 threads. This is lower than the number of chromosomes, 24 (1–22, X, and Y), but some chromosomes (particularly 20–22 and Y) are much smaller than the rest, and in turn, take less time to analyze, so they free one of the threads sooner than the largest ones have finished, thus not affecting negatively speed execution. We did not use 31 threads since assigning more threads than chromosomes does not give an increase in speed. GraffiTE was not included because after assigning it 32 threads, it used 96, which make the results not comparable.

Generation of output files

The output produced by our pipeline includes alignment and variant files produced by minimap2 and the selected variant callers. Additionally, it produces a series of files to facilitate data inspection, visualization and sharing with other researchers.

First, the HTML report includes plots for visualizing AFs, counts for TE variants (both insertions and deletions), presented as tables, also detailing the different types of elements on active subfamilies. TE insertion and deletion counts, gene annotation, and enrichment results. Second, the output includes VCF files with the

inserted sequence, family and subfamily of TE, and AF. One of these files is produced for each of the strict and lax datasets. They do not include identifying information for any sample, for cases in which this would be a concern. All information fields are documented in the header, as the file format requires. Third, if requested, comparison reports are produced for each specified pair of samples with the information shown in "Results". Finally, if enabled, the R objects generated for the analysis are serialized in RDS format, so they can be loaded onto an R session. The contents of the RDS files are explained in RetroInspector's documentation.

Analysis of retrotransposition hallmarks

PALMER[30] was executed on the same three HGSVC samples that were used for benchmarking the pipeline. The results were filtered to keep variants with at least 1 high confidence read and 2 supporting reads (10% of depth coverage, which was rounded down), per its documentation. We also tried thresholds of 3, 5, 7, and 10 high confidence reads. Its results were converted into VCF files with a script that preserves coordinates and length (included in our github repository).

GraffiTE[29] was run following the instructions from its documentation. First, a repeat library was converted to the necessary format. This required to download a complete Dfam release (version 3.8), and then using FamDB toolkit (version 1.0.2, available at https://github.com/Dfam-consortium/FamDB) to extract human (and ancestor) repeats. Second, the same reference genome that was used for RetroInspector and PALMER was selected. Third, "pangenie" was selected as graph method. Finally, the path to the FASTQ files were provided specifying that they are long reads (via the "–longreads" argument) and that they are nanopore reads (by writing "ont" after each entry in the CSV file listing them.

The search of the L1 endonuclease's target and polyA tail was done with the results of the cuteSV-Sniffles2 combination, since it is the one selected as default based on the other benchmarks. For both characteristics it was taken into account that bioinformatic files usually represent one DNA strand, which implies a motif, for example, can be retrieved as its reverse complement.

The consensus motif for the L1's endonuclease nicking target is TTTTT/AA[42] (which may be reported as its reverse complement, TT/AAAAA). Since this is a consensus sequence, a retrotransposition event can occur on similar sequences[43], which were also considered. The reference sequence ±20 bp around the insertion was retrieved, and a fuzzy text search, analogous to a local alignment that allows for substitutions and insertions, for the motif and its reverse complement was done with RapidFuzz[71]. If it returned more than one result, the closest one to the insertion coordinates was selected. Low similarity was allowed at the initial search (a ratio of 60%) with the purpose of later testing higher values, as shown in "Results".

We searched for polyA (or polyT) tails on the reported inserted sequences. To do this, instances of adenine or thymine were recovered. Candidates were then merged into a single candidate if the gap between them was $\leq$ 2bp until all candidates were separated by at least 3 bp. If several results were present, the closest to the end (or the start, to account for the other strand) of the inserted sequence was selected. Finally, the polyA sequence had to start within the last (or first) 10% of the sequence, to differentiate a tail from a low complexity repetition.

For both sequence studies, the significance of differences between true and false positives was checked with a χ-squared test. ROC curves were generated with the R library pROC[72].

The RetroInspector Snakemake workflow

To execute RetroInspector, the users need to provide one FASTQ file per sample (and the path to the directory containing these files), the path to the file with the reference genome, and the output path. Sample names and correspondent files can be written manually or inferred from the FASTQ files. This means that it is possible to perform the whole workflow by specifying just the 3 previous paths. Additionally, the users can set a threshold for the number of supporting reads to filter variants, which pairs of samples to compare, distance margins for merging variants, p-value threshold for enrichment, and the number of threads to utilize. Parameters can be specified at the command line or in a configuration file, following Snakemake best practices. The reference used by default is hg38, which was selected due to the annotation availability. Nonetheless, users may choose to use a different reference, such as T2T[46], and launch the pipeline in "light" mode, which omits gene annotation steps (Fig. 4) that are dependent on the reference version.

It should be noted that the compatibility between the outputs and inputs of consecutive steps has been ensured for all problems detected during development, such as malformed variant records not passing bcftools'[51] assertions.

The Dockerfile for the container version of the pipeline was generated by running `snakemake --containerize> dockerfile`.

PCR validation

PCR primers were designed to only allow for amplification if the insertion was present. We used the sequences reported by nanopore, one primer aligning against the insertion and the other one against the upstream flanking region, which means the amplified sequences, while indicative of the insertion's presence, do not match its full sequence or length. Primers were designed with Python bindings for Primer3[73]. PCRs were done in potential carriers of the inserted TE from our cohort, and in healthy controls from the Spanish general population. An additional blank control was also used in all PCR reactions. Information on the primers is listed in Table S6.

Resource	Reference	URL
Data		
Dfam	Storer et al.[2]	https://www.dfam.org
DO	Schriml[66]	https://disease-ontology.org/
GO	Gene Ontology Consortium[65]	https://geneontology.org/
HGSV nanopore data	Chaisson et al.[31]	ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/data_collections/hgsv_sv_discovery/working/20181210_ONT_rebasecalled/
HGSV2 genotyped variants	Ebert et al.[24]	http://ftp.1000genomes.ebi.ac.uk/vol1/ftp/data_collections/HGSVC2/release/v2.0/integrated_callset/
HGSV2/3 nanopore data	None listed	See accesion numbers in text
Indigen Alu dataset	Prakrithi et al.[35]	https://clingen.igib.res.in/indigen/download/Indigen_Alu_final_geno10_all_22K.vcf
NCG	D'Antonio et al.[67]	http://ncg.kcl.ac.uk/
RepeatMasker output for the human genome	USCS[60]	http://hgdownload.soe.ucsc.edu/goldenPath/hg38/bigZips/hg38.fa.out.gz
Software		
annotatr	Cavalcante and Sartor[62]	https://doi.org/10.18129/B9.bioc.annotatr
BCFtools, bgzip, tabix, and SAMtools	Danecek et al.[51]	https://samtools.github.io/bcftools/bcftools.html
clusterProfiler	Yu et al.[63]	https://doi.org/10.18129/B9.bioc.clusterProfiler
cuteSV	Jiang et al.[52]	https://github.com/tjiangHIT/cuteSV
data.table	Dowle and Srinivasan[68]	https://cran.r-project.org/web/packages/data.table/index.html
DOSE	Yu et al.[64]	https://doi.org/10.18129/B9.bioc.DOSE
minimap2	Li[50]	https://github.com/lh3/minimap2
mosdepth	Pedersen and Quinlan[56]	https://github.com/brentp/mosdepth
pysam	Pysam developers[55]	https://github.com/pysam-developers/pysam
qqman	Turner[69]	https://cran.r-project.org/web/packages/qqman/index.html
R	R core team[61]	https://www.r-project.org/
RepeatMasker	Smit et al.[58]	https://www.repeatmasker.org/
Snakemake	Mölder et al.[49]	https://snakemake.github.io/
Sniffles2	Smolka et al.[19]	https://github.com/fritzsedlazeck/Sniffles
spoa	Vaser et al.[22]	https://github.com/rvaser/spoa
surpyvor	De Coster et al.[70]	https://github.com/wdecoster/surpyvor
SURVIVOR	Jeffares et al.[57]	https://github.com/fritzsedlazeck/SURVIVOR
SVIM	Heller and Vingron[54]	https://github.com/eldariont/svim

Table 6. Data and tools used in this work.

Data availability

The data that support the findings of this study are available from NIHR BioResource but restrictions apply to the availability of these data, which were used under license for the current study, and so are not publicly available (study code DAA067). Requests should be addressed to Javier Corral (javier.corral@carm.es). Public data from other studies has also been used and is cited in Table 6. Data was downloaded from several repositories (Table 6). For HGSV data[31], single FASTQ files were available. For HGSV2[24], files with the following accession numbers were downloaded from SRA[74]: ERX12326582, ERX12326588, ERX12326570, ERX12862034, ERX12862031, ERX12862032, ERX12974695, ERX12326615, ERX12482129, ERX12482121, ERX12482122, ERX12974590, ERX12974494, ERX12482214, ERX12482212, ERX12482215, SRX20457620, ERX12974694, ERX12974699, ERX12974609, ERX12862162, ERX12862163, ERX12862164, ERX12326591, ERX12974551, ERX12862188, ERX12862189, HG03683. RetroInspector is available under the MIT license at https://github.com/javiercguard/retroinspector.

Received: 18 June 2024; Accepted: 15 April 2025
Published online: 25 April 2025

References

1. Kazazian, H. H. & Moran, J. V. Mobile DNA in health and disease. *N. Engl. J. Med.* **377**, 361–370. https://doi.org/10.1056/NEJMra1510092 (2017).
2. Storer, J., Hubley, R., Rosen, J., Wheeler, T. J. & Smit, A. F. The Dfam community resource of transposable element families, sequence models, and genome annotations. *Mobile DNA* **12**, 2. https://doi.org/10.1186/s13100-020-00230-y (2021).
3. Gil, E. et al. Functional characterization of the human mariner transposon Hsmar2. *PLoS One* **8**, e73227. https://doi.org/10.1371/journal.pone.0073227 (2013).
4. Ayarpadikannan, S. & Kim, H.-S. The impact of transposable elements in genome evolution and genetic instability and their implications in various diseases. *Genom. Inform.* **12**, 98–104. https://doi.org/10.5808/GI.2014.12.3.98 (2014).
5. Yu, L. et al. New insights into the evolution of intronic sequences of the beta-fibrinogen gene and their application in reconstructing mustelid phylogeny. *Zool. Sci.* **25**, 662–672. https://doi.org/10.2108/zsj.25.662 (2008).

6. Hung, T. et al. The Ro60 autoantigen binds endogenous retroelements and regulates inflammatory gene expression. *Science* **350**, 455–459. https://doi.org/10.1126/science.aac7442 (2015).
7. Norris, J. et al. Identification of a new subclass of Alu DNA repeats which can function as estrogen receptor-dependent transcriptional enhancers. *J. Biol. Chem.* **270**, 22777–22782. https://doi.org/10.1074/jbc.270.39.22777 (1995).
8. Hancks, D. C. & Kazazian, H. SVA retrotransposons: Evolution and genetic instability. *Semin. Cancer Biol.* **20**, 234–245. https://doi.org/10.1016/j.semcancer.2010.04.001 (2010).
9. Zhou, S. & Van Bortle, K. The Pol III transcriptome: Basic features, recurrent patterns, and emerging roles in cancer. *Wiley Interdiscip. Rev. RNA* **14**, e1782. https://doi.org/10.1002/wrna.1782 (2023).
10. Stenz, L. The L1-dependant and Pol III transcribed Alu retrotransposon, from its discovery to innate immunity. *Mol. Biol. Rep.* **48**, 2775–2789. https://doi.org/10.1007/s11033-021-06258-4 (2021).
11. Burns, K. H. Our conflict with transposable elements and its implications for human disease. *Annu. Rev. Pathol.* **15**, 51–70. https://doi.org/10.1146/annurev-pathmechdis-012419-032633 (2020).
12. Bouras, A. et al. Identification and characterization of new Alu element insertion in the BRCA1 exon 14 associated with hereditary breast and ovarian cancer. *Genes* **12**, 1736. https://doi.org/10.3390/genes12111736 (2021).
13. Reiter, L. T. et al. A recombination hotspot responsible for two inherited peripheral neuropathies is located near a mariner transposon-like element. *Nat. Genet.* **12**, 288–297. https://doi.org/10.1038/ng0396-288 (1996).
14. Prakrithi, P. et al. An Alu insertion map of the Indian population: Identification and analysis in 1021 genomes of the IndiGen project. *NAR Genom. Bioinform.* **4**, lqac009. https://doi.org/10.1093/nargab/lqac009 (2022).
15. Chu, C. et al. Comprehensive identification of transposable element insertions using multiple sequencing technologies. *Nat. Commun.* **12**, 3836. https://doi.org/10.1038/s41467-021-24041-8 (2021).
16. Logsdon, G. A., Vollger, M. R. & Eichler, E. E. Long-read human genome sequencing and its applications. *Nat. Rev. Genet.* **21**, 597–614. https://doi.org/10.1038/s41576-020-0236-x (2020).
17. Vendrell-Mir, P. et al. A benchmark of transposon insertion detection tools using real data. *Mobile DNA* **10**, 53 (2019).
18. Puurand, T., Kukuškina, V., Pajuste, F.-D. & Remm, M. AluMine: Alignment-free method for the discovery of polymorphic Alu element insertions. *Mobile DNA* **10**, 31. https://doi.org/10.1186/s13100-019-0174-3 (2019).
19. Smolka, M. et al. Detection of mosaic and population-level structural variants with Sniffles2. *Nat. Biotechnol.* https://doi.org/10.1038/s41587-023-02024-y (2024).
20. Cuenca-Guardiola, J. et al. Detection and annotation of transposable element insertions and deletions on the human genome using nanopore sequencing. *iScience* **26**. https://doi.org/10.1016/j.isci.2023.108214 (2023).
21. Lee, H., Min, J. W., Mun, S. & Han, K. Human retrotransposons and effective computational detection methods for next-generation sequencing data. *Life* **12**, 1583. https://doi.org/10.3390/life12101583 (2022).
22. Vaser, R., Sović, I., Nagarajan, N. & Šikić, M. Fast and accurate de novo genome assembly from long uncorrected reads. *Genome Res.* **27**, 737–746. https://doi.org/10.1101/gr.214270.116 (2017).
23. Sahlin, K. & Medvedev, P. Error correction enables use of Oxford Nanopore technology for reference-free transcriptome analysis. *Nat. Commun.* **12**, 2. https://doi.org/10.1038/s41467-020-20340-8 (2021).
24. Ebert, P. et al. Haplotype-resolved diverse human genomes and integrated analysis of structural variation. *Science (New York, N.Y.)* **372**, eabf7117. https://doi.org/10.1126/science.abf7117 (2021).
25. Zabolotneva, A. A. et al. Transcriptional regulation of human-specific $SVAF_1$ retrotransposons by cis-regulatory MAST2 sequences. *Gene* **505**, 128–136. https://doi.org/10.1016/j.gene.2012.05.016 (2012).
26. Chu, C. et al. The landscape of human SVA retrotransposons. *Nucleic Acids Res.* **51**, 11453–11465. https://doi.org/10.1093/nar/gkad821 (2023).
27. Mohamed, M. et al. TrEMOLO: Accurate transposable element allele frequency estimation using long-read sequencing data combining assembly and mapping-based approaches. *Genome Biol.* **24**, 63. https://doi.org/10.1186/s13059-023-02911-2 (2023).
28. Charron, P. & Kang, M. VariantDetective: An accurate all-in-one pipeline for detecting consensus bacterial SNPs and SVs. *Bioinformatics* btae066. https://doi.org/10.1093/bioinformatics/btae066 (2024).
29. Groza, C., Chen, X., Wheeler, T. J., Bourque, G. & Goubert, C. A unified framework to analyze transposable element insertion polymorphisms using graph genomes. *Nat. Commun.* **15**, 8915. https://doi.org/10.1038/s41467-024-53294-2 (2024).
30. McDonald, T. L. et al. Cas9 targeted enrichment of mobile elements using nanopore sequencing. *Nat. Commun.* **12**, 3586. https://doi.org/10.1038/s41467-021-23918-y (2021).
31. Chaisson, M. J. P. et al. Multi-platform discovery of haplotype-resolved structural variation in human genomes. *Nat. Commun.* **10**, 1784. https://doi.org/10.1038/s41467-018-08148-z (2019).
32. HGSV Consortium. HGSV nanopore data. ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/data_collections/hgsv_sv_discovery/working/20181210_ONT_rebasecalled
33. de la Morena-Barrio, B. et al. Long-read sequencing identifies the first retrotransposon insertion and resolves structural variants causing antithrombin deficiency. *Thromb. Haemost.* **122**, 1369–1378. https://doi.org/10.1055/s-0042-1749345 (2022).
34. Graham, T. & Boissinot, S. The genomic distribution of L1 elements: The role of insertion bias and natural selection. *J. Biomed. Biotechnol.* **2006**, 75327. https://doi.org/10.1155/JBB/2006/75327 (2006).
35. Prakrithi, P., Singhal, K., Sharma, D. et al. Indigen Alu Final Geno10 All 22K VCF File. *Clingen* https://clingen.igib.res.in/indigen/download/Indigen_Alu_final_geno10_all_22K.vcf (2022).
36. Gorbunova, V. et al. The role of retrotransposable elements in ageing and age-associated diseases. *Nature* **596**, 43–53. https://doi.org/10.1038/s41586-021-03542-y (2021).
37. Saleh, A., Macia, A. & Muotri, A. R. Transposable elements, inflammation, and neurological disease. *Front. Neurol.* **10**, 894. https://doi.org/10.3389/fneur.2019.00894 (2019).
38. Deininger, P. Alu elements: Know the SINEs. *Genome Biol.* **12**, 236. https://doi.org/10.1186/gb-2011-12-12-236 (2011).
39. Cardelli, M. The epigenetic alterations of endogenous retroelements in aging. *Mech. Ageing Dev.* **174**, 30–46. https://doi.org/10.1016/j.mad.2018.02.002 (2018).
40. Chen, J. et al. Alu methylation serves as a biomarker for non-invasive diagnosis of glioma. *Oncotarget* **7**, 26099–26106. https://doi.org/10.18632/oncotarget.8318 (2016).
41. Thawani, A., Ariza, A. J. F., Nogales, E. & Collins, K. Template and target site recognition by human LINE-1 in retrotransposition. *Nature* https://doi.org/10.1038/s41586-023-06933-5 (2023).
42. Thawani, A., Ariza, A. J. F., Nogales, E. & Collins, K. Template and target-site recognition by human LINE-1 in retrotransposition. *Nature* **626**, 186–193. https://doi.org/10.1038/s41586-023-06933-5 (2024).
43. Feng, Q., Moran, J. V., Kazazian, H. H. & Boeke, J. D. Human L1 retrotransposon encodes a conserved endonuclease required for retrotransposition. *Cell* **87**, 905–916. https://doi.org/10.1016/S0092-8674(00)81997-2 (1996) (**Publisher: Elsevier.**).
44. Bilgrav Saether, K. & Eisfeldt, J. Detecting transposable elements in long-read genomes using sTELLeR. *Bioinformatics* **40**, btae686. https://doi.org/10.1093/bioinformatics/btae686 (2024).
45. Shestak, A. G., Bukaeva, A. A., Saber, S. & Zaklyazminskaya, E. V. Allelic dropout is a common phenomenon that reduces the diagnostic yield of PCR-based sequencing of targeted gene panels. *Front. Genet.* **12**, 620337. https://doi.org/10.3389/fgene.2021.620337 (2021).
46. Aganezov, S. et al. A complete reference genome improves analysis of human genetic variation. *Science (New York, N.Y.)* **376**, eabl3533. https://doi.org/10.1126/science.abl3533 (2022).

47. Gardner, E. J. et al. The Mobile Element Locator Tool (MELT): Population-scale mobile element discovery and biology. *Genome Res.* **27**, 1916–1929. https://doi.org/10.1101/gr.218032.116 (2017).
48. Yu, Q. et al. Population-wide sampling of retrotransposon insertion polymorphisms using deep sequencing and efficient detection. *GigaScience* **6**, 1–11. https://doi.org/10.1093/gigascience/gix066 (2017).
49. Mölder, F. et al. Sustainable data analysis with Snakemake. Tech. Rep. 10:33. *F1000Res.*. https://doi.org/10.12688/f1000research.29032.2 (2021).
50. Li, H. Minimap2: Pairwise alignment for nucleotide sequences. *Bioinformatics* **34**, 3094–3100. https://doi.org/10.1093/bioinformatics/bty191 (2018).
51. Danecek, P. et al. Twelve years of SAMtools and BCFtools. *GigaScience* **10**, giab008. https://doi.org/10.1093/gigascience/giab008 (2021).
52. Jiang, T. et al. Long-read-based human genomic structural variation detection with cuteSV. *Genome Biol.* **21**, 189. https://doi.org/10.1186/s13059-020-02107-y (2020).
53. Jiang, T. et al. Long-read sequencing settings for efficient structural variation detection based on comprehensive evaluation. *BMC Bioinform.* **22**, 552. https://doi.org/10.1186/s12859-021-04422-y (2021).
54. Heller, D. & Vingron, M. SVIM: Structural variant identification using mapped long reads. *Bioinformatics* **35**, 2907–2915. https://doi.org/10.1093/bioinformatics/btz041 (2019).
55. Pysam developers. Pysam (2021).
56. Pedersen, B. S. & Quinlan, A. R. Mosdepth: Quick coverage calculation for genomes and exomes. *Bioinformatics* **34**, 867–868. https://doi.org/10.1093/bioinformatics/btx699 (2018).
57. Jeffares, D. C. et al. Transient structural variations have strong effects on quantitative traits and reproductive isolation in fission yeast. *Nat. Commun.* **8**, 14061. https://doi.org/10.1038/ncomms14061 (2017).
58. Smit, A., Hubley, R. & Green, P. RepeatMasker Open-4.0 (2013).
59. Camacho, C. et al. BLAST+: Architecture and applications. *BMC Bioinform.* **10**, 421. https://doi.org/10.1186/1471-2105-10-421 (2009).
60. UCSC. RepeatMasker .out file. *UCSC*. http://hgdownload.soe.ucsc.edu/goldenPath/hg38/bigZips/hg38.fa.out.gz (2014).
61. R Core Team. *R: A Language and Environment for Statistical Computing* (R Foundation for Statistical Computing, 2020).
62. Cavalcante, R. G. & Sartor, M. A. annotatr: Genomic regions in context. *Bioinformatics* **33**, 2381–2383. https://doi.org/10.1093/bioinformatics/btx183 (2017).
63. Yu, G., Wang, L.-G., Han, Y. & He, Q.-Y. clusterProfiler: An R package for comparing biological themes among gene clusters. *OMICS J. Integr. Biol.* **16**, 284–287. https://doi.org/10.1089/omi.2011.0118 (2012).
64. Yu, G., Wang, L.-G., Yan, G.-R. & He, Q.-Y. DOSE: An R/Bioconductor package for disease ontology semantic and enrichment analysis. *Bioinformatics (Oxford, England)* **31**, 608–609. https://doi.org/10.1093/bioinformatics/btu684 (2015).
65. The Gene Ontology Consortium et al. The Gene Ontology resource: enriching a GOld mine. *Nucleic Acids Res.* **49**, D325–D334. https://doi.org/10.1093/nar/gkaa1113 (2021).
66. Schriml, L. M. et al. Human Disease Ontology 2018 update: Classification, content and workflow expansion. *Nucleic Acids Res.* **47**, D955–D962. https://doi.org/10.1093/nar/gky1032 (2019).
67. D'Antonio, M., Pendino, V., Sinha, S. & Ciccarelli, F. D. Network of Cancer Genes (NCG 3.0): integration and analysis of genetic and network properties of cancer genes. *Nucleic Acids Res.* **40**, D978–D983. https://doi.org/10.1093/nar/gkr952 (2012).
68. Dowle, M. & Srinivasan, A. *data.table: Extension of 'data.frame'* (2021).
69. Turner, D.S. qqman: An R package for visualizing GWAS results using Q-Q and Manhattan plots. *J. Open Source Softw.* **3**, 731. https://doi.org/10.21105/joss.00731 (2018).
70. De Coster, W. et al. Structural variants identified by Oxford Nanopore PromethION sequencing of the human genome. *Genome Res.* **29**, 1178–1187. https://doi.org/10.1101/gr.244939.118 (2019).
71. Bachmann, M. rapidfuzz/rapidfuzz: Release 3.8.1. https://doi.org/10.5281/zenodo.10938887 (2024).
72. Robin, X. et al. proc: An open-source package for r and s+ to analyze and compare roc curves. *BMC Bioinform.* **12**, 77 (2011).
73. Untergasser, A. et al. Primer3-new capabilities and interfaces. *Nucleic Acids Res.* **40**, e115. https://doi.org/10.1093/nar/gks596 (2012).
74. Katz, K. et al. The Sequence Read Archive: A decade more of explosive growth. *Nucleic Acids Res.* **50**, D387–D390. https://doi.org/10.1093/nar/gkab1053 (2022).

Author contributions
J.C.G implemented the pipeline. J.C.G, B.M.B, J.C. and J.T.F.B designed the pipeline, conceived the experiment(s) and wrote the manuscript. All authors reviewed the manuscript.

Funding
Javier Cuenca-Guardiola is funded by the Ministerio de Universidades through grant FPU19/03662. This work has been possible thanks to the funding of the Instituto de Salud Carlos III (ISCIII) through the projects "PI21/00174" and "PMP21/00052", which were co-funded by the European Union and by the European Union - Next Generation EU, respectively. Belén de la Morena has a postdoctoral contract from CIBERER.

Declarations

Competing interests
The authors declare no competing interests.

Ethical approval
The study protocol was approved by the Clinical Research Ethics Committee at Morales Meseguer University Hospital (code EST: 31/18) and conducted in accordance with the 1964 Declaration of Helsinki and their later amendments. All included subjects gave their written informed consent to enter the study.

Additional information
Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-025-98847-7.

Correspondence and requests for materials should be addressed to J.T.F.-B.