OXFORD (GIGA)n SCIENCE

GigaScience, 2024, **13**, 1–15
DOI: 10.1093/gigascience/giae040
Technical Note

Deciphering cancer genomes with GenomeSpy: a grammar-based visualization toolkit

Kari Lavikka [1,*], Jaana Oikkonen [1], Yilin Li [1], Taru Muranen [1], Giulia Micoli [1], Giovanni Marchi [1], Alexandra Lahtinen [1], Kaisa Huhtinen [1,2], Rainer Lehtonen [3], Sakari Hietanen [4], Johanna Hynninen [4], Anni Virtanen [5], and Sampsa Hautaniemi [1,*]

[1]Research Program in Systems Oncology, Research Programs Unit, Faculty of Medicine, University of Helsinki, 00014 Helsinki, Finland
[2]Cancer Research Unit, Institute of Biomedicine and FICAN West Cancer Centre, University of Turku, 20521 Turku, Finland
[3]Applied Tumor Genomics Research Program, Research Programs Unit, Faculty of Medicine, University of Helsinki, 00014 Helsinki, Finland
[4]Department of Obstetrics and Gynecology, University of Turku and Turku University Hospital, 20521 Turku, Finland
[5]Department of Pathology, University of Helsinki and HUS Diagnostic Center, Helsinki University Hospital, 00260 Helsinki, Finland
*Correspondence address. Kari Lavikka, Research Program in Systems Oncology, Research Programs Unit, Faculty of Medicine, University of Helsinki, 00014 Helsinki, Finland, E-mail: kari.lavikka@helsinki.fi; Sampsa Hautaniemi, Research Program in Systems Oncology, Research Programs Unit, Faculty of Medicine, University of Helsinki, 00014 Helsinki, Finland, E-mail: sampsa.hautaniemi@helsinki.fi

Abstract

Background: Visualization is an indispensable facet of genomic data analysis. Despite the abundance of specialized visualization tools, there remains a distinct need for tailored solutions. However, their implementation typically requires extensive programming expertise from bioinformaticians and software developers, especially when building interactive applications. Toolkits based on visualization grammars offer a more accessible, declarative way to author new visualizations. Yet, current grammar-based solutions fall short in adequately supporting the interactive analysis of large datasets with extensive sample collections, a pivotal task often encountered in cancer research.

Findings: We present GenomeSpy, a grammar-based toolkit for authoring tailored, interactive visualizations for genomic data analysis. By using combinatorial building blocks and a declarative language, users can implement new visualization designs easily and embed them in web pages or end-user–oriented applications. A distinctive element of GenomeSpy's architecture is its effective use of the graphics processing unit in all rendering, enabling a high frame rate and smoothly animated interactions, such as navigation within a genome. We demonstrate the utility of GenomeSpy by characterizing the genomic landscape of 753 ovarian cancer samples from patients in the DECIDER clinical trial. Our results expand the understanding of the genomic architecture in ovarian cancer, particularly the diversity of chromosomal instability.

Conclusions: GenomeSpy is a visualization toolkit applicable to a wide range of tasks pertinent to genome analysis. It offers high flexibility and exceptional performance in interactive analysis. The toolkit is open source with an MIT license, implemented in JavaScript, and available at https://genomespy.app/.

Keywords: genomic data visualization, visualization grammar, GPU-accelerated visualization, ovarian high-grade serous carcinoma

Introduction

Effective visualization facilitates hypothesis generation and the assessment of automatic analyses, making it an indispensable facet of genomic data analysis [1]. However, interpreting complex genomic datasets calls for visualization methods tailored to the analyzed data [2], a need underscored by the availability of numerous special-purpose tools [3, 4]. Implementing such tailored visualizations, particularly those that offer interactivity, presents a significant challenge for most researchers [5]. It typically necessitates developing new software packages from scratch using low-level libraries such as D3 [6] or writing plugins for existing ones like the modular JBrowse 2 [7] genome browser. This laborious process demands considerable programming expertise beyond most bioinformaticians' skills.

Visualization grammars like ggplot2 [8], Vega-Lite [9], and the genomic data–focused Gosling [10] and ggbio [11], which all build upon the concept initially presented in the Grammar of Graphics [12], support tailored visualizations with a more accessible approach: instead of using an imperative programming language, they are specified using combinatorial building blocks such as graphical marks, scales, transformations, and view compositions, which are put together using a declarative language. However, none of these grammar-based solutions sufficiently cater to the typical analysis task in cancer research: the exploration and analysis of large sample collections to find patterns and outliers in cohorts. They lack support for genomic data, fail to visualize numerous concurrent samples, disallow interactive filtering and grouping, or underperform with large datasets.

Herein, we present GenomeSpy, a toolkit designed to simplify the crafting of interactive visualizations and empower end users to effectively explore and analyze large datasets, particularly in cancer research. The toolkit features a grammar that enables effortless implementation of different visualization strategies (Fig. 1). This characteristic makes GenomeSpy fundamentally distinct from genome browsers, such as IGV [13], igv.js [14], JBrowse 2, and UCSC Genome Browser [15], which comprise predefined track

Received: January 9, 2024. **Revised:** May 13, 2024. **Accepted:** June 19, 2024

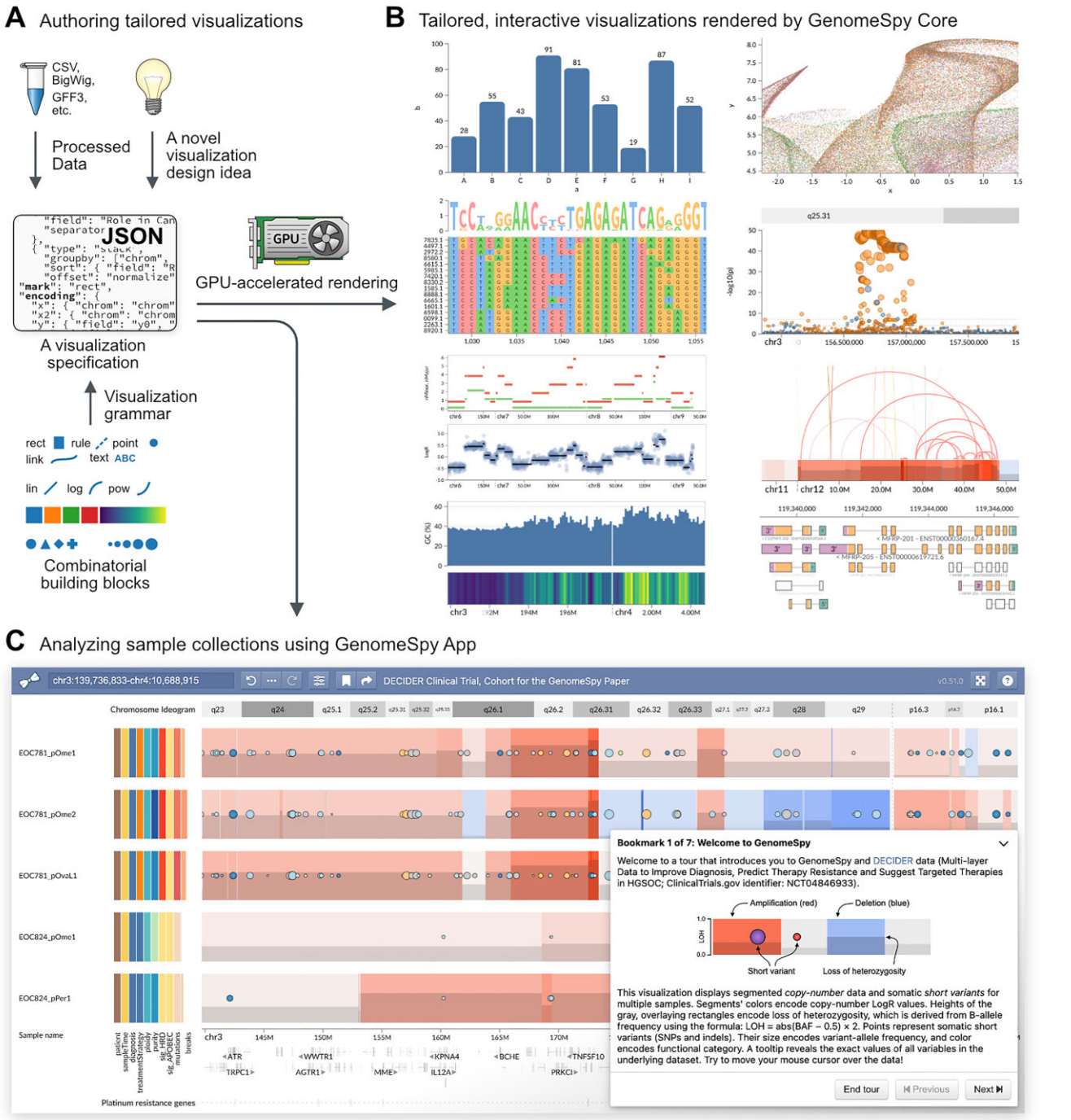


Figure 1: Overview of GenomeSpy. (A) GenomeSpy enables tailored visualizations through its JSON-based visualization grammar, which defines how the building blocks, such as marks and scales, can be combined into a visualization specification. Instead of relying on predefined templates or track types, the user can freely compose visualizations from various graphical marks and map data attributes to different visual channels, such as color and position. (B) GenomeSpy core library parses the specification and renders it using GPU-accelerated graphics to ensure smooth interactions such as zooming and panning. Interactive versions of the above examples are available at https://genomespy.app/ [17]. (C) GenomeSpy App builds upon the core and enables the analysis of sample collections. The above visualization with 753 samples is available for exploration at https://csbi.ltdk.helsinki.fi/p/genomespy-paper-2024/ [18].

types designed for specific data formats that are displayed using rigid visual encodings. In addition, we incorporated the grammar into an analysis application for sample collections, with a pronounced focus on fluid interaction. This design principle aims to make interaction with visualizations more rewarding, ultimately enhancing users' performance [16]. Fluid interaction changes browsing and exploration, which are considered a rate-limiting step in data analysis [2], into an endeavor that fosters insights.

We demonstrate the utility and key features of GenomeSpy by exploring and analyzing 753 whole-genome sequencing (WGS) samples from 215 patients who belong to prospective, longitudinal, multiregion observational study DECIDER (Multi-layer Data to Improve Diagnosis, Predict Therapy Resistance and Suggest Targeted Therapies in HGSOC; ClinicalTrials.gov identifier: NCT04846933) that started recruitment in 2012. The DECIDER trial focuses on characterizing and overcoming therapy resistance in

ovarian high-grade serous carcinoma (HGSC), the most common and aggressive epithelial ovarian cancer subtype. The standard of care (SOC) for HGSC consists of debulking surgery and platinum-taxane chemotherapy, often combined with maintenance therapy with ADP ribose polymerase (PARP) or vascular endothelial growth factor (VEGF) pathway inhibitors [19]. While ∼80% of HGSC patients respond well to the SOC, most of the patients have recurrence and rapid disease progression, leading to a 5-year survival rate of only <40% [20]. Except for nearly 100% prevalent TP53 mutations, HGSC lacks recurrent mutations but is characterized by complex genomes with large-scale copy number alterations, hindering a deeper mechanistic understanding of the disease [21, 22]. Furthermore, diagnosis is often complicated by rare morphologic and molecular traits [23, 24]. Herein, our hypothesis is that interpreting large genomics datasets from genomically complex cancers, such as HGSC, requires tailored visualization methods, such as one built with GenomeSpy.

Results

GenomeSpy is a JavaScript-based toolkit that allows developers and bioinformaticians to build interactive visualizations for genome analysis. To construct such a visualization, a user writes a visualization specification in JavaScript Object Notation (JSON) format, adhering to the rules of the visualization grammar (Fig. 1A). GenomeSpy's grammar draws inspiration from the design principles of Vega-Lite, a high-level grammar of interactive graphics [9], enhancing it for robust support of genomic data (Supplementary Note). Fig. 2 demonstrates GenomeSpy's grammar-based approach with a typical use case: a nucleotide sequence of a reference genome.

The *core library* constitutes the toolkit's main component. It implements the grammar and renders the visualization according to the provided specification (Fig. 1B). The library can serve as a component in JavaScript web applications, or it can be embedded on webpages such as Observable notebooks (https://observablehq.com/collection/@tuner/genomespy [25]). An example of a special-purpose application built using the core library is SegmentModel Spy (Fig. 3, Supplementary Note), which allows a comprehensive assessment of copy number segmentation output from the Genome Analysis Toolkit (GATK) [26]. A crucial element in the core library's architecture is its use of graphics processing unit (GPU) acceleration through the WebGL 2 API for all graphics and scale transformations (Supplementary Note). GPU acceleration enables efficient rendering with a high frame rate and minimal latency, which facilitates insight generation [27]. It also allows fluid, smoothly animated interactions, such as continuous zooming and panning in large datasets. While smooth transition animations make the visualizations more attractive, they have also been shown to improve users' perception of causality during interactions [28].

The *app* is a general-purpose analytics application for large sample collections, built upon the core library (Fig. 1C). It permits interactive analysis of genomic data and metadata, such as clinical variables. Using the grammar, users can adapt the app for different data types and analysis tasks. The app allows storing its state in the form of bookmarks or shareable links. The state comprises current scale domains (i.e., shown genomic regions and the visibility of configurable visualization elements). The state also captures the filtering, grouping, and sorting actions performed on the samples, serving as provenance information that allows the recipient of a shared bookmark link to understand which steps led to a finding or insight [30, 31]. Finally, bookmarks also support optional Markdown-formatted notes, which allow communicating background information or implications related to the findings.

The *playground* web application (https://genomespy.app/playground/ [32]) integrates a code editor and a visualization, providing a convenient way to sketch new visualization designs. It is also the easiest method for new users to get started with GenomeSpy.

In addition to a specification, GenomeSpy visualizations need data, which can be provided as inline JavaScript objects in the specification or loaded from external files. CSV, TSV, and JSON files provide the highest flexibility. However, large datasets are better loaded lazily and only partially in response to user interactions, which is supported through compressed and indexed formats, such as BigBed, BigWig, FASTA, and GFF3 files. Additionally, GenomeSpy's JavaScript API provides methods to dynamically update the datasets, enabling advanced use cases, such as integrations with and within other applications, as shown in Fig. 3. The loaded data can be further processed using GenomeSpy's built-in data transformation pipeline, which is fully configurable through the visualization grammar. The transformation steps also support parameterization, allowing the pipeline's behavior to be changed interactively using sliders and other user-interface controls, thus enabling a deeper level of data exploration beyond basic interactions such as zooming in and out or hovering with tooltips. Furthermore, GenomeSpy's client-side data-processing model offers developers the convenience of a simplified setup that obviates the need for specialized server-side infrastructure.

Characterizing the genomic landscape of HGSC

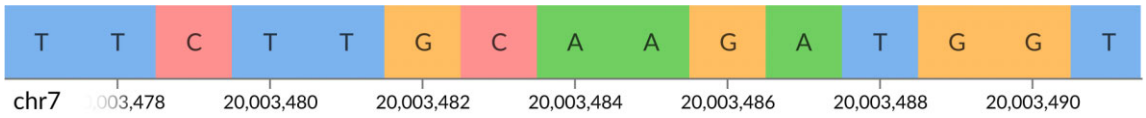
We demonstrate the utility of the toolkit and highlight GenomeSpy App's key features by showing how they enable the interpretation of WGS data from 753 samples of 215 patients belonging to the DECIDER clinical trial. Using GenomeSpy's visualization grammar, we adapted the app for our data by specifying a visualization comprising segmented copy number alterations (CNAs), loss of heterozygosity (LOH), somatic short variants (SSVs), and clinical data as shown in Fig. 1C. We also specified several tracks exhibiting auxiliary information, such as ENCODE Blacklist [33], RefSeq Gene annotations [34], and genes associated with platinum resistance [35]. Some of these tracks are hidden by default and can be activated from the toolbar. The visualization is available for exploration at https://csbi.ltdk.helsinki.fi/p/genomespy-paper-2024/ [18], and for those wishing to adapt it for their own datasets, we provide the specification with documentation and example data at https://github.com/HautaniemiLab/genomespy-paper-2024-spec [36].

Rapid transitions between the bird's-eye view and a close-up facilitate exploration

To streamline the exploration of large sample collections, we developed an interaction that rapidly transits the visualization from the bird's-eye view, which fits the whole collection into the available vertical space, to a close-up view, where the samples under the mouse cursor are shown in a larger size (Supplementary Video). This interaction allows for pinpointing interesting outliers among hundreds of samples and rapidly revealing them in sufficient detail for visual analysis, streamlining the exploration process. GenomeSpy's GPU-accelerated rendering is pivotal in this feature, as it guarantees smooth transition between the views.

We used the bird's-eye view to gain an overview of the cohort. While recurrent *TP53* mutations and LOH on chromosome 17 (chr17) are known genomic aberrations in HGSC and contribute to

A



B

```
{
  "genome": { "name": "hg38" },

  "data": {                                          (1)
    "lazy": {
      "type": "indexedFasta",
      "url": "https://data.genomespy.app/genomes/hg38/hg38.fa"
    }
  },

  "transform": [                                     (2)
    { "type": "flattenSequence", "field": "sequence", "as": ["rawPos", "base"] },
    { "type": "formula", "expr": "datum.rawPos + datum.start", "as": "pos" }
  ],

  "encoding": {                                      (3)
    "x": { "chrom": "chrom", "pos": "pos", "type": "locus" },
    "color": { "field": "base", "type": "nominal" }
  },

  "layer": [                                         (4)
    { "mark": "rect" },                              (5)
    {
      "mark":  "text",
      "encoding": {
        "color": { "value": "black" },
        "text": { "field": "base" }
      }
    }
  ]
}
```

Figure 2: Specifying a visualization of a reference nucleotide sequence using the grammar. (A) The example visualization comprises letters that are superimposed on colored rectangles. The genomic axis is generated automatically. (B) The GenomeSpy core library provides no predefined track types. Instead, the visualization author supplies a JSON-based specification that defines how the building blocks are put together. (1) The *data* property specifies a data source. In this example, data are loaded lazily from an indexed FASTA file as the user navigates the genome. (2) Optional *transformations* modify the data stream. Here, the sequence strings provided by the data source are split into data objects representing individual nucleotides with their coordinates. (3) The *encoding* property allows mapping data fields to visual channels. The x-axis is treated as genomic coordinates, as it has a "locus" data type. (4) The *layer* property composes multiple child views by layering them. (5) The *mark* property specifies the graphical mark to be used in a view. Here, "rect" is used for the background rectangles and "text" for the bases. N.B. The specification has been simplified for clarity by omitting noncritical properties. A complete example is available in GenomeSpy's documentation.

tumor evolution [22, 37, 38], the concurrent display of both copy number values and LOH revealed a striking pattern in the bird's-eye view: regardless of copy number gains and losses in chr17, all but 5 patients presented a complete LOH in the whole chromosome (Fig. 4). The whole-chromosome LOH suggests an early mitotic nondisjunction affecting the entire chromosome, with subsequent alterations, such as 17q amplifications, arising at a later stage.

We then looked more closely at the outliers that had retained chr17 heterozygosity by opening the close-up view (Supplementary Video). Three of these outliers lacked a *TP53* mutation, which is atypical in HGSC. Thus, a gynecological pathologist reevaluated these cases, and the diagnoses of patients EOC466 and EOC545 were changed to low-grade serous carcinoma (LGSC) and EOC571 to endometrioid carcinoma. One of the outliers had lost heterozygosity only on 17p and was subsequently found to present endometroid carcinoma. The only HGSC tumor without chr17 LOH stood out with a massive number of somatic mutations, indicating a possible mismatch-repair deficiency, which is a hallmark of Lynch syndrome. As Lynch syndrome results from germline mutations in DNA mismatch repair genes, we examined them and found a germline mutation in *MSH6*, which accounts for 10–20% of Lynch syndromes in colorectal cancer [40]. Since Lynch syndrome is dominantly inherited, these results were reported to a clinical geneticist to be discussed with the patient's family.

Incremental, reversible actions enable rapid manipulation of the sample collection

Data exploration often involves the removal of irrelevant data items or organizing the data to uncover patterns. To facilitate this process, we developed a direct manipulation interface [41] that allows for incremental actions on abstract attributes such as clinical metadata or measurements at genomic loci. These actions can be accessed through a context menu (Fig. 5), permitting the user to easily perform common tasks such as retaining samples belonging to a particular categorical class or stratifying samples based on a quantitative value at a specific genomic coordinate. Additionally, the actions are reversible, allowing for backtracking and further exploration of related questions. The actions also form a

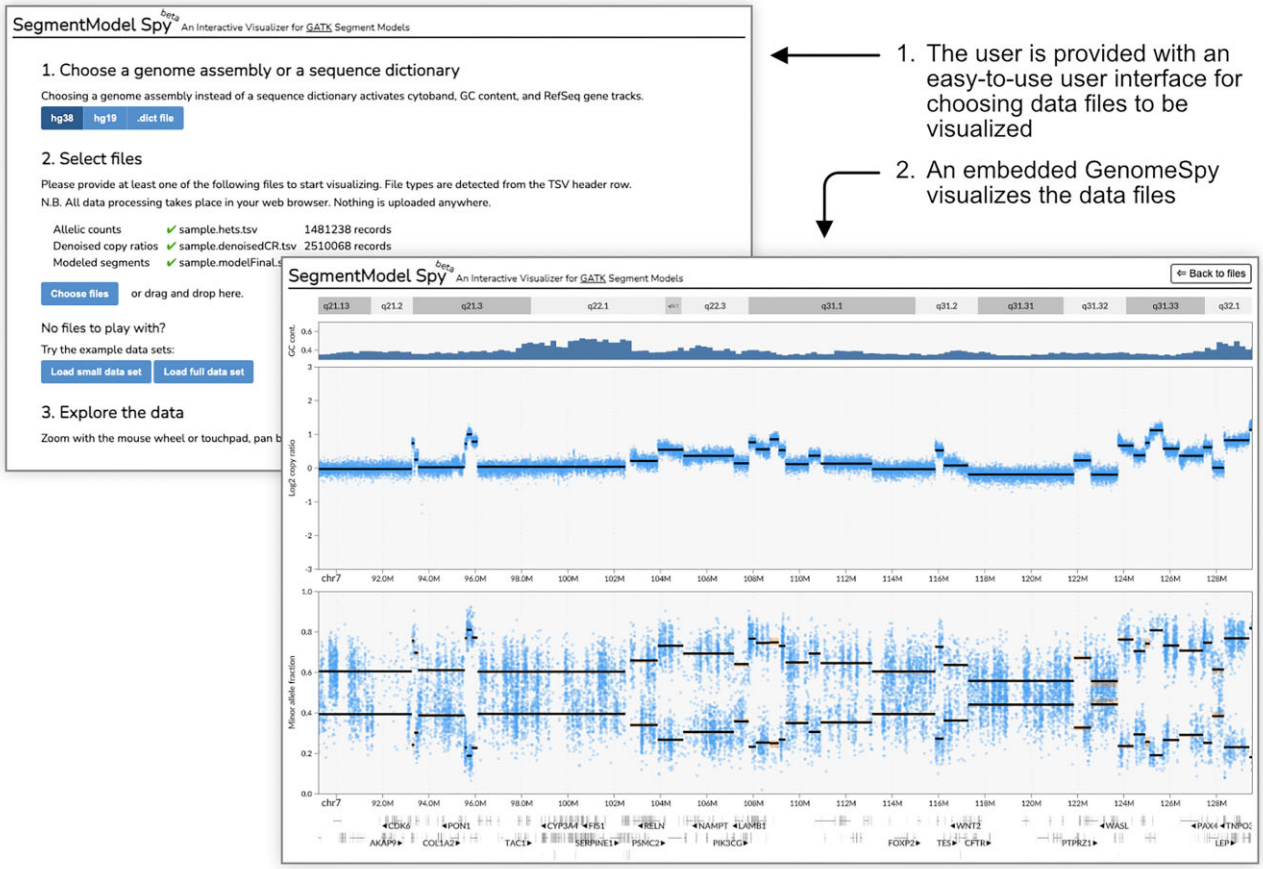


Figure 3: SegmentModel Spy demonstrates GenomeSpy's utility as a visualization library in JavaScript web applications. It is a simple, end-user–oriented application for analyzing GATK's copy number segmentation results, allowing users to open data files effortlessly for swift navigation and inspection. The application generates a visualization specification and passes it with the parsed data files to the embedded GenomeSpy core library for visualization. Notably, all data processing occurs in the user's web browser without the involvement of a remote server, enabling the analysis of sensitive data. SegmentModel Spy is available with example data at https://genomespy.app/segmentmodel/ [29].

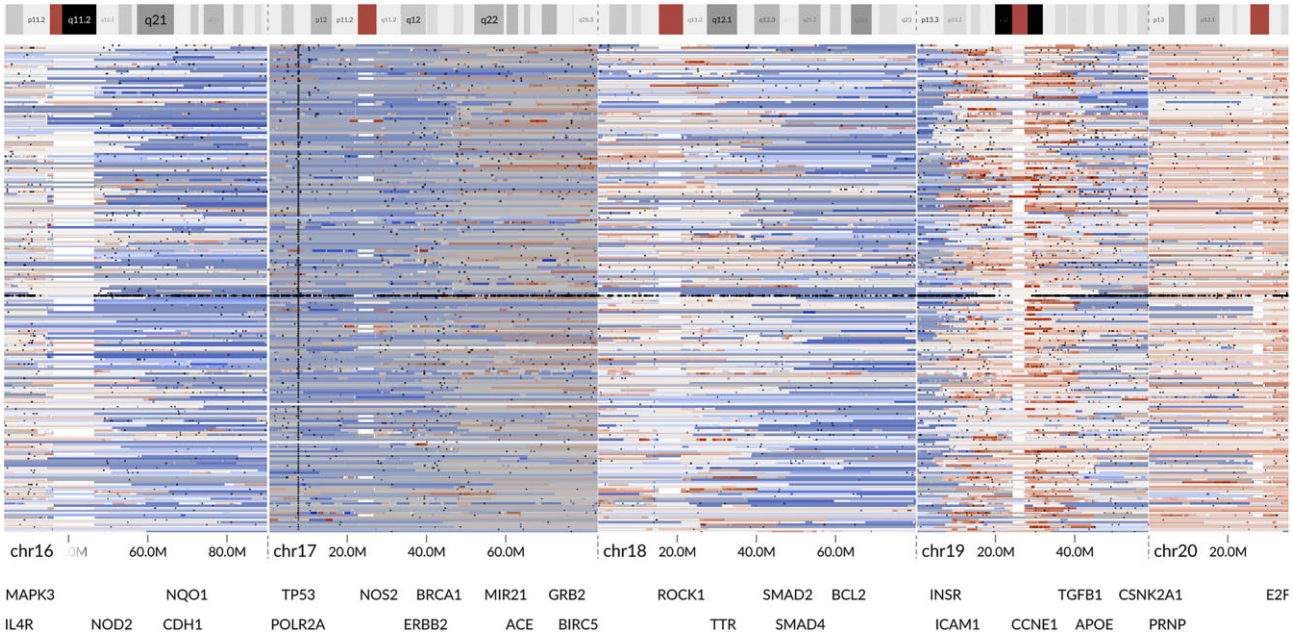


Figure 4: A bird's-eye view of all patients reveals a column of *TP53* mutations (dark dots) together with extensive LOH (gray overlay) on chr17. Only the sample with the highest purity (at least 15%) is included from each patient. One of the samples presents a very high number of SSVs and retained chr17 heterozygosity. The remaining 4 samples without full-chromosome LOH were from non-HGSC tumors. Interactive visualization: [39].

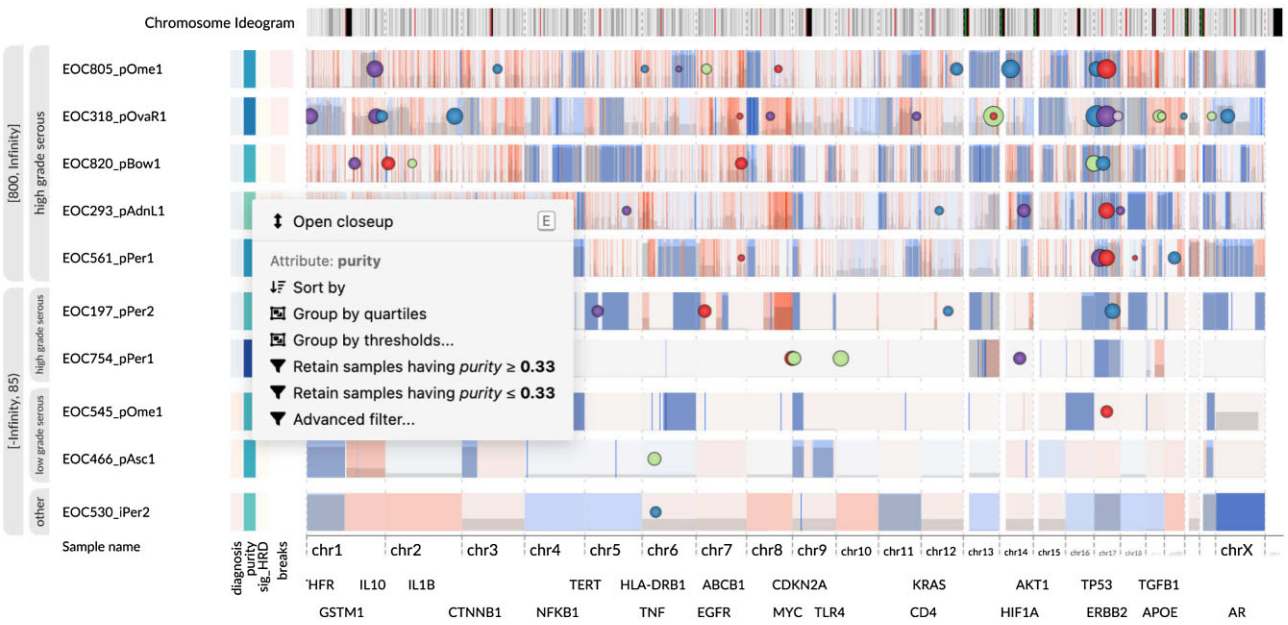


Figure 5: Top and bottom 5 samples by the number of copy number breakpoints. Only the sample with the highest number of breakpoints was chosen from each patient. A nested, second-level grouping emphasizes the diagnosis attribute. The upper group exhibits a striking pattern of short amplifications associated with *CDK12* inactivation. The bottom group contains 3 samples from non-HGSC tumors and a peculiar HGSC tumor sample (EOC754_pPer1) with very few CNAs. The view was constructed using the incremental actions available through the attribute context menu (shown in the screenshot). Interactive visualization: [42].

provenance record of the steps taken in the data exploration process, ensuring transparency and reproducibility.

HGSC is characterized by extensive copy number aberrations [22]. However, we observed considerable variation in the number of copy number breakpoints between the patients. To better understand this variation, we applied a series of incremental actions to shape and stratify our sample set. First, we selected samples having purity at least 15%. We then sorted the samples into descending order by the number of breakpoints and retained the first, representative sample from each patient, which corresponded to the most fragmented one. Finally, we split the samples into groups based on the number of breakpoints and analyzed the patients with the most and least fragmented tumor genomes (Fig. 5).

The 5 most highly fragmented samples showed a striking pattern of numerous focal amplifications evenly distributed throughout the genome. These amplifications ranged in size from ∼100 kb to ∼10 Mb. The zoomed-out whole-genome view also revealed deleterious (stop-gain or frameshift) *CDK12* SSVs (visible in chr17 in the figure) in 4 of the 5 samples. The allele frequencies of the variants matched the tumor purity, suggesting homozygous mutations and thus biallelic inactivation. Of note, 4 of these 5 samples presented copy-neutral LOH in the *CDK12* locus, suggesting subsequent amplification after the initial chr17 loss. Previous research has linked *CDK12* inactivation to a specific type of chromosomal instability characterized by tandem duplications with a bimodal size distribution, which is in line with our observation [43]. Interestingly, when visualizing all samples from these patients (Fig. 6), the amplification pattern is nearly identical among the samples of each patient, implying subsequent stabilization of the genomes.

Next, we focused on the 5 patients with the fewest breakpoints. Two of them (EOC466 and EOC545) were previously found to have LGSC based on the lack of a *TP53* mutation. Additionally, patient EOC530, who also lacked a *TP53* mutation but still exhibited chr17 LOH, had a nonserous neoplasm diagnosis. The 2 remaining patients had an HGSC diagnosis, but EOC754's tumor presented a peculiar copy number profile with aberrations only in 3 chromosomes. Although the mutated *TP53* and chr17 LOH in this tumor were consistent with the histological diagnosis of HGSC, the copy number profile was surprising since it had even fewer arm-level aberrations than the 2 samples from LGSC patients.

We further analyzed the cohort for mitogen-activated protein kinase (MAPK) pathway genes commonly altered in LGSC [45] and found *NRAS*:c.182A>G:p.Q61R in the samples from EOC530, EOC545, and EOC754, as well as another oncogenic aberration *BRAF*:c.1862A>G:p.N621S in samples from EOC466. Otherwise, oncogenic *NRAS* mutations were not detected in the entire cohort, and *BRAF* mutations were present in only 2 additional patients with characteristically simple copy number profiles. Generally, *NRAS* mutations are rarely seen in HGSC carcinomas but more commonly in borderline or low-grade serous neoplasms [46], as seen in patient EOC545.

As patient EOC754 exhibited an *NRAS* mutation and an atypical copy number profile resembling the low-grade serous carcinomas of patients EOC545 and EOC466, a gynecological pathologist performed a retrospective histological review of her archival tumor samples. The tumor had a serous phenotype, but in terms of histological architecture, cytological atypia, and mitotic rate, the tumor, especially in ovarian samples, showed areas with unequivocally low-grade morphology in addition to areas with more pronounced pleiomorphism and mitotic activity. Yet, all 4 samples with sequencing data from this patient showed LOH on the whole chromosome 17 and a clonal *TP53* mutation in addition to *NRAS*. Cases with such genomic and morphological features from both high- and low-grade serous carcinomas have previously been reported as rare variants of serous ovarian neoplasms [24, 47]. A further study on the potential origin and genomic and histological evolution of this and the 2 *BRAF*-mutated HGSC cases discovered through exploration in GenomeSpy is ongoing.

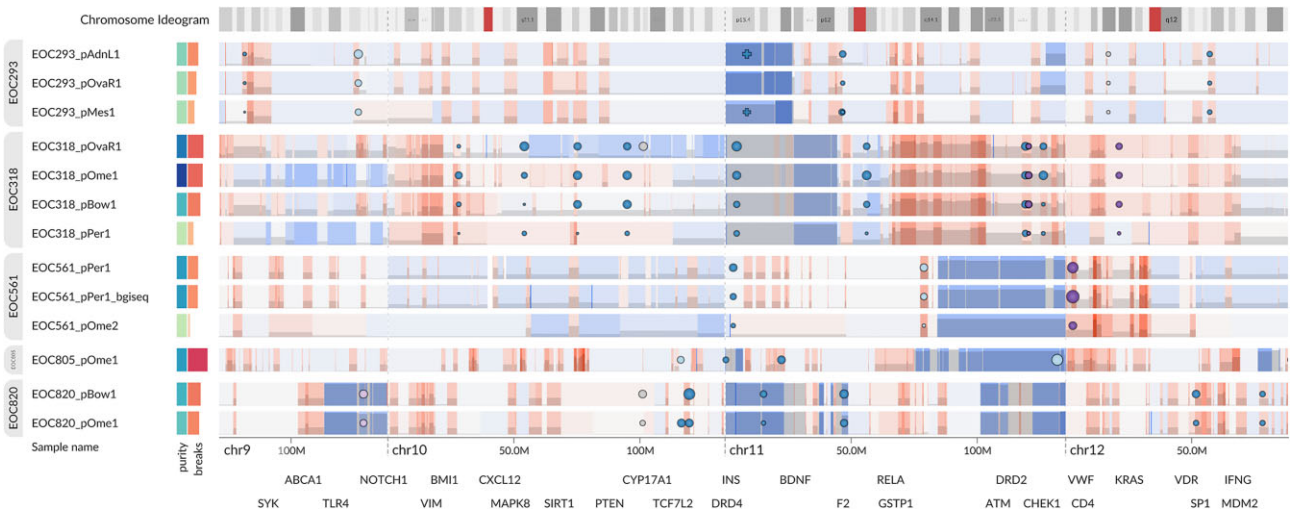


Figure 6: The amplified segments associated with the tandem-duplication phenotype and *CDK12* inactivation are largely identical within the samples of each patient, suggesting subsequent genome stabilization. Samples with a very low tumor purity, which are indicated by light green in the metadata heatmap, suffer from low segmentation sensitivity and lack some of the segments that were detected in high-purity samples. Interactive visualization: [44].

Score-based semantic zoom emphasizes important data items and mitigates overplotting

While somatic mutations are one of the driving forces behind tumorigenesis, most of the detected SSVs are passengers without contribution to disease. However, they clutter the view, making prompt identification of the pathogenic driver SSVs challenging. On the other hand, displaying all SSVs at once may be advantageous when an analyst studies a small genomic region that may accommodate SSVs with still uncertain pathogenicity. To address these conflicting needs, we developed *score-based semantic zoom*, a technique that couples a filter on an arbitrarily distributed quantitative attribute (i.e., a score) with the zoom level (Supplementary Note, Supplementary Video). In the zoomed-out view, only the most important (i.e., the highest scored data points) are shown, allowing the user to locate potentially important data items for a closer examination. As the user zooms in, items with lower scores become visible automatically, without the need to adjust separate filter settings. This behavior resembles online map applications where only the largest and most well-known place names are initially visible, with more names appearing gradually as the map is zoomed in. This technique also helps to avoid overplotting by controlling the number of concurrently visible data items.

To facilitate analysis and control overplotting, we applied the semantic zoom technique to all SSVs in the dataset. For scoring, we used the Combined Annotation-Dependent Depletion (CADD) score [48], a single measure that integrates a diverse set of annotations. Thus, only the most likely pathogenic variants are shown at each zoom level. For instance, the recurrent *TP53* mutations and the *CDK12* mutations linked to chromosomal instability are visible already in the fully zoomed-out view (Fig. 5), while the lower-scored variants remain out of sight until the user zooms in closer. This feature allowed us to instantly discover the pathogenic *CDK12* SSVs in the highly fragmented samples.

Data summarization allows easier comparison of stratified data

Although a CNA heatmap presents all details in data, a summary, such as the GISTIC G-score [49], enables an easier perception of potential cancer driver regions and facilitates the comparison of groups. Accordingly, we used GenomeSpy's visualization grammar to specify a summary track that computes G-scores over the segmented copy number data. The summary incorporates a pipeline of transformations that inputs the copy number values from the currently visible samples and computes a weighted coverage separately for amplifications and deletions (see Methods). The user can interactively adjust various threshold parameters to focus better on broad or focal regions. A summary of the highest purity samples from all HGSC patients revealed a typical HGSC CNA landscape with prominent peaks around common HGSC driver genes [22], such as *MECOM*, *MYC*, *KRAS*, and *CCNE1* (Fig. 7).

Next, we asked whether the recurrent amplification and deletion peaks in HGSC could be explained by clinical attributes or correlation of potential driver regions. Because the G-score summary track reflects the currently visible samples and is computed separately for each group, we could easily analyze stratified data by visually comparing the G-scores. However, stratifications failed to reveal distinguishable differences with attributes other than tumor ploidy. When we stratified the tumors into 2 groups approximately representing whole-genome duplicated (WGD) and non-WGD tumors, an evident focal amplification peak around *CCNE1* in chr19 was present only in the WGD group, as shown in Fig. 8. Previous research has associated such *CCNE1* amplifications with polyploidy and poor clinical outcome [51].

Data visualization helps in finding clinically actionable alterations

With the increased efforts to guide treatment decisions based on genomics findings, there is a need to rapidly visualize genomes to verify findings and detect aberrations that were not caught with automatic data analysis pipelines. For example, *BRCA1* is a tumor suppressor gene that contributes to DNA repair, and its mutation is an indication for targeted therapy with PARP inhibitors in HGSC [19].

As the PARP inhibitors are currently the only genomic-guided targeted therapy in HGSC, we used GenomeSpy to visually inspect the loci of *BRCA1* and other homologous recombination deficiency-related genes in our samples and identified a suspi-

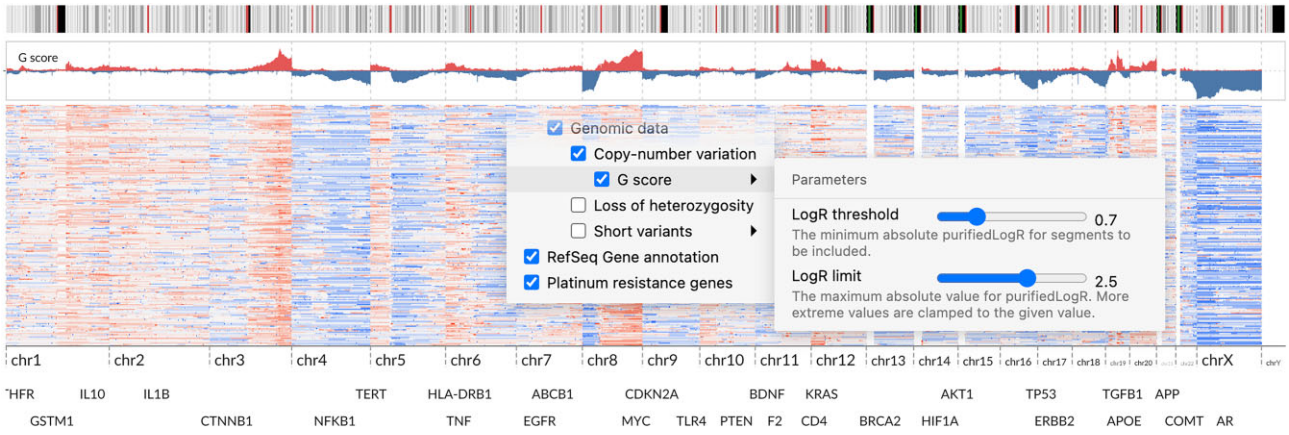


Figure 7: Using G-score to summarize the copy number landscape of the cohort. It is shown as an area chart above the copy number heatmap. We used building blocks such as sample summarization, various transformations, and view compositions in the visualization specification to calculate and display the G-score. Some parameters in the transformations are bound to input controls, allowing the user to adjust the calculation interactively. Interactive visualization: [50].

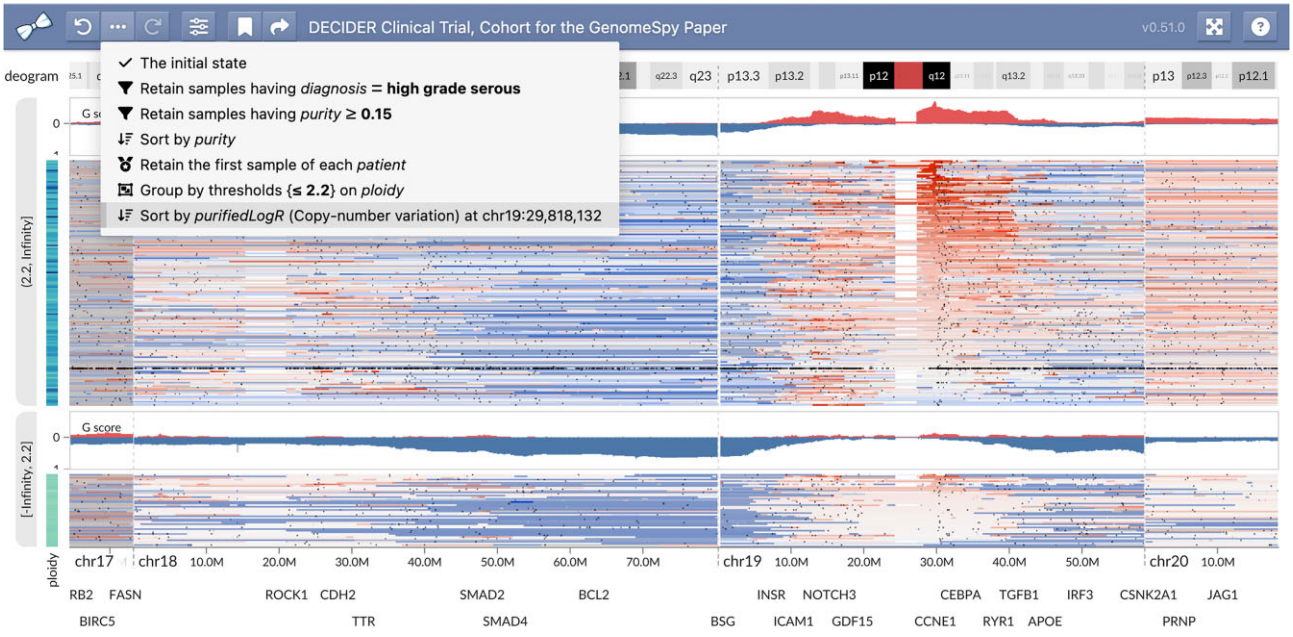


Figure 8: HGSC samples stratified by ploidy revealed higher *CCNE1* amplifications (shown in red) in the upper group that represents whole-genome-duplicated samples. Each group has a separately computed G-score summary to allow comparison. The opened drop-down menu reveals the provenance information comprising actions performed on the samples. Since most patients have samples from multiple tissues and time points, we kept only the highest-purity sample of each patient. Subsequently, we used the ploidy threshold of 2.2 to split the samples into 2 groups approximately representing non-WGD and WGD. Finally, we sorted the samples by the copy number of *CCNE1* to better illustrate the distributions of the copy number $\log_2$(R) values in both groups. Interactive visualization: [52].

cious *BRCA1* region for one patient. A copy number pipeline, which employs GRIDSS [53] for joint structural variation calling, confirmed a multiexon in-frame deletion of *BRCA1* (chr17:43,096,222–43108182del, p.(K45_S198delinsN)) in all sequenced tumor samples from this patient (Fig. 9). The deletion comprised exons 4–8, covering half of the RING domain. With supporting information from mutation signature analysis and the known consequences of similar medium-long deletions of *BRCA1* in ClinVar [54], we interpreted this *BRCA1* allele as pathogenic. Accordingly, the finding enabled the use of a PARP inhibitor to treat the patient in a recurrent setting. This example highlights the potential of visualization methods, such as GenomeSpy, in searching for genomically based treatments for cancer patients.

Discussion

Visual exploration is a necessary step in oncogenomic data analysis and knowledge extraction [56]. To facilitate the exploration, we developed GenomeSpy, a visualization toolkit for genomic data. Two main objectives steered the process: designing a generic toolkit that enables effortless authoring of tailored visualizations for different use cases and implementing a fully customizable application to analyze large cancer sample collections. We achieved the genericity by implementing a grammar optimized for genomic data and demonstrated its expressivity (i.e., its applicability to complex data), using the DECIDER cohort visualization. To support the swift analysis of sample collections, we applied the paradigm of fluid interaction [16], which manifested as sev-

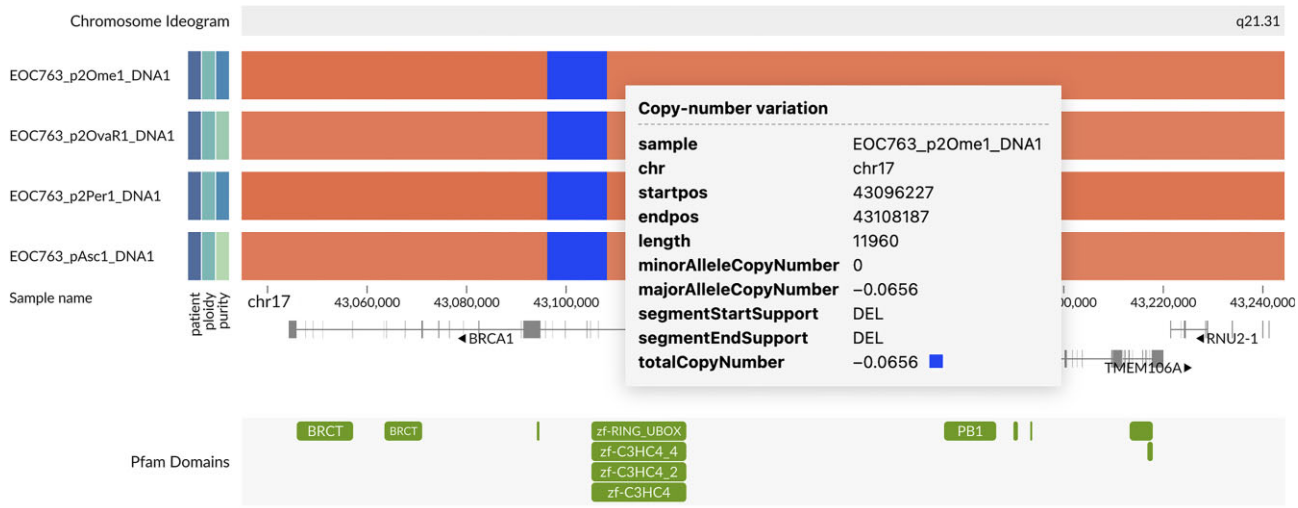


Figure 9: Results from an experimental copy number pipeline revealed a homozygous *BRCA1* deletion in all tumor samples of patient EOC763. Because the pipeline could not directly output $\log_2(R)$ and BAF values, we used the total copy number instead of $\log_2(R)$ on the color channel of this visualization. Interactive visualization: [55].

eral key features and influenced the overall architecture of the toolkit. For instance, we designed a GPU-accelerated rendering engine to allow rapidly updating graphics with an extensive number of data points. In addition to supporting continuous zooming and panning, it enabled the interaction that transits between the bird's-eye view and a close-up view, allowing quick examination of outliers in datasets. Importantly, its smooth animation helps the user stay focused without losing track of data not currently on the screen. Similarly, the score-based semantic zoom controls overplotting during navigation, allowing the user to focus on the most important data items at each zoom level. Finally, the direct-manipulation interface [41] based on incremental actions enables quick and versatile stratification and exploration with support for backtracking, bookmarks, and provenance information. All these features aim to expedite the exploration and thus foster insights.

GenomeSpy allowed us to characterize the genomic landscape of the DECIDER cohort, uncovering several interesting patterns. Among our findings, the extent and completeness of the chr17 LOH was one of the most surprising. Although such LOH has been previously found to occur in some ovarian carcinoma tumors [38], our dataset shows that out of the 200 representative HGSC tumor samples having a purity of at least 15%, all but 1, which had multiple *TP53* mutations, presented whole-chromosome LOH on chr17. While the whole-chromosome LOH allows a nascent tumor to expunge the remaining wild-type *TP53*, the same mechanism may also contribute to the biallelic inactivation or reduced dosage of other tumor suppressor genes in the same chromosome, such as *CDK12*, *BRCA1*, *BRIP1*, and *NF1* [23, 57]. This hypothesis is supported by the pathogenic *CDK12* mutations associated with the tandem-duplicator phenotype. Homozygosity coupled with the copy-neutral LOH in these mutations indicates early occurrence, before or soon after the whole-chromosome loss. In addition to cohort characterization, GenomeSpy also enabled the discovery of exciting outliers, such as the tumor of one patient with traits from both HGSC and LGSC. Overall, the effective use of visual encodings and the high usability provided by fluid interaction have established GenomeSpy as an indispensable analysis tool among our geneticists, especially with copy number data, whose interpretation requires a view of the larger genomic context. Moreover, an example of using GenomeSpy to facilitate the identification of genomic-based personalized treatment is the discovery of an actionable *BRCA1* deletion, which was not detected with a commercial panel, most likely due to the small size of the deletion.

GenomeSpy visualizations allow end users, such as geneticists, clinicians, and bioinformaticians, to analyze datasets with ease. While our grammar-based approach simplifies the creation of such interactive visualizations considerably, there is still an acknowledged learning curve similar to what has been observed with other visualization toolkits [5, 58]. However, the formal JSON-based grammar opens future opportunities for easy point-and-click visualization designer interfaces [59], visualization recommendation systems [60], and more streamlined grammar versions [61] to further ease the authoring process. On the other hand, because many analysis tasks already have established visualization designs and techniques, we aim to support them through predefined templates, much like the standard track types in genome browsers. This will enhance immediate usability for newcomers and lower the initial learning barrier. Moreover, as the Segment-Model Spy example highlighted (Fig. 3), the toolkit can be used to build easy-to-use applications for specific analysis tasks. Expanding on this, we envision GenomeSpy as a foundation for a next-generation general-purpose genome browser that provides a comprehensive collection of datasets and predefined track types powered by extensive customizability and high-performance interactivity.

Conclusions

In conclusion, we have demonstrated GenomeSpy's flexibility and utility with the visualization of a cohort from the DECIDER clinical trial, and we envision the toolkit as a foundation for many future applications. The grammar-based design of GenomeSpy enables a modular approach, where the various building blocks of visualization—graphical marks, scales, transformations, and compositions—can be combined and reconfigured in innovative ways to meet specific research needs. GenomeSpy is open-source, welcoming contributions and advancements, with full documentation at https://genomespy.app/ [17].

Materials and Methods

GenomeSpy core

The core library is written in JavaScript. It uses the WebGL API and the TWGL library [62] for GPU-accelerated graphics. In addition, D3 [6] and Vega [63] libraries are used for CPU-side scale transformations, data loading, and expression handling. Genomic file formats, such as indexed FASTA, BigWig, BigBed, and GFF3, are loaded using GMOD JavaScript libraries [7]. The core library is available as an NPM package, which can be imported into web applications, webpages, and Observable notebooks. A more detailed description of the architecture and visualization grammar is available in the Supplementary Note and the GenomeSpy website [17].

GenomeSpy App

The app builds upon the core library. It uses the Redux Toolkit [64] for state management and provenance tracking and Lit [65] for user-interface components. The application is available as an NPM package, which can be embedded on webpages together with a visualization specification and data.

DECIDER cohort

Multi-layer Data to Improve Diagnosis, Predict Therapy Resistance and Suggest Targeted Therapies in HGSOC (DECIDER; ClinicalTrials.gov identifier: NCT04846933) is a prospective, longitudinal, multiregion observational study that began recruitment in 2012. Herein, we included 215 patients treated at Turku University Hospital, Finland. The treatment was either primary debulking surgery (PDS), followed by a median of 6 cycles of platinum-taxane chemotherapy, or neoadjuvant chemotherapy (NACT), where primary laparoscopic operation with diagnostic tumor sampling was followed by 3 cycles of carboplatin and paclitaxel.

Altogether, we included all 753 tumor samples that had been whole-genome sequenced when the cohort was formed. The samples comprise tumor tissue (tubo-ovarian, intra-abdominal, and other metastatic sites such as lymph nodes) and ascites from several phases of the disease.

All patients participating in the study gave their informed consent. The study and the use of all clinical materials have been approved by the Ethics Committee of the Hospital District of Southwest Finland under decision number VARHA/28314/13.02.02/2023.

Whole-genome sequencing

Genomic DNA was extracted from tumor tissue or ascites cells and whole blood or buffy coats isolated from whole blood. After assessing DNA quality, the samples were whole-genome sequenced with DNBSEQ (BGISEQ-500 or MGISEQ-2000; MGI Tech Co., Ltd.), Illumina NovaSeq 6000 (RRID:SCR_016387), or Illumina HiSeq X Ten (RRID:SCR_016385) as 150-bp paired-end sequencing. Median coverage was ~47× (range, 23×–158×). Raw read data were processed with Trimmomatic (RRID:SCR_011848) [66] and FastQC (RRID:SCR_014583) [67] in the Anduril 2 workflow platform [68]. The reads were then aligned to the human genome GRCh38.d1.vd1 using BWA-MEM (RRID:SCR_022192), followed by a duplicate removal with Picard (RRID:SCR_006525) [69] and base quality score [70] recalibration with GATK (RRID:SCR_001876) [71].

Mutation calling

We called somatic mutations using GATK *Mutect2* [72] with joint calling [73]. A panel of normals generated from 181 DECIDER and 99 TCGA blood-derived normal samples was used. Mutations were annotated using ANNOVAR (RRID:SCR_012821) [74], ClinVar [54], and CADD (RRID:SCR_018393) estimates for deleteriousness [48]. Germline mutations were jointly called using GATK [73] from 217 DECIDER normal samples with allele-specific variant quality score recalibration. Variant quality score recalibration was allele specific. Mutational signatures were fitted using COSMIC v3.2 signatures [75], adjusted for GRCh38 nucleotide frequencies.

Copy number calling and estimation of ploidy and tumor purity

We used GATK to perform the copy number segmentation. The analysis pipeline follows the GATK best-practices documentation and builds upon the Anduril 2 platform.

To collect the minor allele counts (b-allele frequency, BAF), we used all filtered biallelic germline single-nucleotide polymorphisms (SNPs) with heterozygous calls (VAF between 40% and 60%) from each patient. Both read and allelic count collection excluded regions listed in the ENCODE blacklist [33] and our internal DECIDER blacklist, which is available as a track in the DECIDER visualization. The DECIDER blacklist includes regions having $\text{abs}(\log_2(R)) > 0.2$ in at least 3 of the 114 normal samples used as input data. The 136 regions in the DECIDER blacklist represent poorly aligned regions and population-level copy number variance. We used platform-specific (HiSeq, DNBSEQ, and NovaSeq) panels of normals built from the normal samples to denoise the read counts.

Since the result of the actual segmentation affects downstream analyses such as ploidy and purity estimation, we visually evaluated the effect of the various parameters of GATK's *ModelSegments* tool. In practice, we ran the segmentation for select samples using 729 different combination of values for the parameters and studied their effect using the SegmentModel Spy tool (Fig. 3, Supplementary Note). Finally, we chose parameters that resulted in the subjectively best breakpoint inference results. For instance, short segments should be included, but false breakpoints related to GC-wave artifacts need to be avoided. The final parameters were as follows: number-of-changepoints-penalty-factor: 1, kernel-variance-allele-fraction: 0, kernel-variance-copy-ratio: 0.2, kernel-scaling-allele-fraction: 0.1, smoothing-credible-interval-threshold-allele-fraction: 2, smoothing-credible-interval-threshold-copy-ratio: 10.

After the segmentation, we used a reimplemented ASCAT algorithm [76] to estimate purity, ploidy, and allele-specific copy numbers. The original ASCAT R package [77] was not directly applicable because it fails to accept data segmented using external tools. Our implementation also uses the variant allele frequency (VAF) of truncal pathogenic *TP53* mutation as additional evidence in selection of the optimal ploidy/purity solution. As nearly all patients have a homozygous *TP53* mutation in their cancer cells, we can use the VAF and the estimated total copy number (CN) of *TP53* to approximate the purity:

$$\text{purity}_{TP53} = 2/\left((\text{CN}_{TP53}/\text{VAF}_{TP53}) - (\text{CN}_{TP53} - 2)\right).$$

Patients having discordant ploidy estimates between their samples went through manual curation.

Since the contribution of nonaberrant cells on the $\log_2(R)$ and BAF values encumber visualization and further analyses, we calculated "purified" values, that is, what the $\log_2(R)$ would be in the absence of normal cells.

Purified R, based on discussion in [78]:

$$\text{purifiedR} = (\text{purity} \times \text{ploidy} \times \text{R} + 2 \times (1 - \text{purity}) \times (\text{R} - 1)) / (\text{purity} \times \text{ploidy})$$

Purified BAF, derived from S2, S7, and S8 of [77]:

$$\text{f}(\text{af}) = \text{purity} - 1 + \text{R} \times \text{af} \times (2 \times (1 - \text{purity}) + \text{purity} \times \text{ploidy})$$
$$\text{purifiedBaf} = \text{f}(\text{baf}) / (\text{f}(1 - \text{baf}) + \text{f}(\text{baf}))$$

Experimental copy number pipeline for *BRCA1/2* analysis

We called structural variants in a callset of 139 DECIDER patients using GRIDSS [53] with joint calling and performed the somatic filtering using GRIPSS [79] with a panel of normals from Dutch population [80] and the ENCODE blacklist [33]. The BAF was calculated using AMBER [81] with the heterozygous SNP loci from the mutation calling. Read depth was extracted using COBALT [82], which also performed GC normalization. Finally, we employed PURPLE [80] to combine BAF, read depth ratios, and structural variants to estimate purity, ploidy, and the copy number profile of the samples.

Pathogenic *BRCA1/2* mutations

We curated somatic and germline short variants in *BRCA1/2* genes. We considered a variant pathogenic if it causes premature truncation in the canonical transcript or if it is annotated as pathogenic or likely pathogenic in the ClinVar [54] database. For patient homozygosity assessment, we compared allelic read counts against allele-specific copy numbers in the locus and purities in tumor samples with a minimum purity of 5%. A variant was considered homozygous if it was the most likely explanation for the allelic read counts across a patient's tumor samples.

DECIDER cohort visualization

We used the GenomeSpy app for the DECIDER visualization. Annotation tracks such as RefSeq genes are specified in separate JSON files, allowing easy reuse. The main JSON file specifies the visualization of metadata, SSVs, CNV, BAF, and the copy number summary. GenomeSpy inputs all genomic and metadata from tab-separated value (TSV) files.

Only SSVs with the CADD score of at least 10.0 or that were pathogenic according to ClinVar [54] were included to reduce loading time and memory consumption. We used the purified $\log_2$(R) and BAF values for CNV and LOH, allowing more meaningful comparison, sorting, and grouping. To enable easier perception of aberrant BAF, we converted it into LOH using the formula:

$$\text{LOH} = \text{abs}(\text{BAF} - 0.5) \times 2.$$

Here, 0 indicates full heterozygosity, and 1 indicates a total loss of heterozygosity.

The dynamically updating copy number summary track replicates the G-score of GISTIC 1.0. The initial GISTIC version was chosen because its G-score formula is straightforward to implement with GenomeSpy's grammar. It also allows much quicker interaction speeds for real-time analysis, unlike the more complex GISTIC 2.0 [83]. Briefly, the dataflow processes amplifications and deletions separately. Only segments with abs(purifiedLogR) > 0.7 are included and abs(purifiedLogR) is clamped to 2.5. These 2 thresholds can be adjusted interactively by the user. Finally, the dataflow computes a purifiedLogR-weighted coverage for the segments and divides it by the number of samples involved. Coverages of amplifications and deletions have separate layers in the visualization and are shown as red and blue, respectively.

The RefSeq gene annotation track uses a popularity-based prioritization for the gene symbols [84], a method originally introduced in HiGlass [85]. Thus, at each zoom level, the symbols are handled in priority order and shown if there is still room on the track.

Availability of Source Code and Requirements

Project name: GenomeSpy
Project homepage: https://genomespy.app/ [17] and https://github.com/genome-spy/genome-spy [86]
Operating systems: Platform independent
Programming languages: JavaScript and TypeScript
License: MIT
RRID:SCR_024,837

Additional Files

Supplementary Fig. S1. An example of a visualization specification with 2 layers rendered in GenomeSpy (top) and Vega-Lite (bottom). GenomeSpy adapts the design of Vega-Lite's grammar, providing partial compatibility. However, the implementation is independent and makes extensive use of GPU in scale transformations and rendering. In this example, the dataset is embedded into the specification and comprises objects with 2 fields: a and b. The *encoding* block specifies how the data fields are mapped to different visual channels. In this case, the a field is declared as nominal data and mapped onto the x-axis, and the y field is quantitative and mapped onto the y-axis. The *layer* block specifies 2 superimposed graphical marks: *rect* forms the bars on the chart and *text* shows the exact data values above the bars.

Supplementary Fig. S2. Handling genomic data in GenomeSpy using transformations and scales. (A) In an abstract sense, a transformation inputs a list of data items and outputs a list of new items that may be filtered, modified, or generated from the original items. GenomeSpy provides a *linearizeGenomicCoordinate* transformation that maps the discrete chromosomes onto a single linear coordinate space. (B) By using the *locus* data type and specifying the *chrom* and *pos* fields, an implicit linearization transformation is added to the data flow, allowing easy handling of genomic coordinates. (C) The *locus* scale maps the linearized genomic coordinates to the viewport and provides chromosome-aware axis ticks. (D) GenomeSpy provides several data transformations that enable visualization techniques commonly used with genomic data. For example, when working with overlapping segments, the *coverage* transformation (upper plot) generates a list of new segments with continuous coverage values and the *pileup* transformation (lower plot) assigns each segment a free lane. Both plots use the *rect* mark to visualize the transformed data items.

Supplementary Fig. S3. Chrome Developer Tools profile of GenomeSpy when zooming the whole dataset (753 samples) in the visualization shown in Fig. 1C (https://csbi.ltdk.helsinki.fi/p/genomespy-paper-2024). The profile shows very low CPU utilization, which guarantees smooth animation without dropped frames.

Supplementary Fig. S4: Chrome Developer Tools profile of Gosling (v0.12.0) when zooming the Visual Encoding example (https://gosling.js.org/?example=VISUAL_ENCODING). The profile shows

full CPU utilization and multiple dropped frames, resulting in choppy animation during interactions.

Supplementary Table S1. Comparison of Gosling and GenomeSpy.

Supplementary Note. A note covering the following topics: Comparison to Vega-Lite, The GPU-accelerated Architecture, Embedding in Web Applications: Segmentmodel Spy, Exploring Sample Collections with the GenomeSpy App, and Comparison to Gosling.

Supplementary Video. A video demonstrating two GenomeSpy features: Toggling between the bird's eye view and a closeup, Score-based semantic zoom applied to somatic short variants.

Data Availability

The following resources have been deposited in the Software Heritage Archive: the source code of the GenomeSpy toolkit [89], SegmentModel Spy [90], and the reimplemented ASCAT algorithm [76]. The DECIDER HGSC visualization specification with documentation and example data has been archived in Zenodo [36]. All sequencing data are available at the European Genome-phenome Archive (EGA) under accession number EGAS00001006775.

Abbreviations

BAF: B-allele frequency; CNV: copy number variance; DECIDER: Multi-layer Data to Improve Diagnosis, Predict Therapy Resistance and Suggest Targeted Therapies in HGSOC; GPU: graphics processing unit; JSON: JavaScript Object Notation; LOH: loss of heterozygosity; PARP: ADP ribose polymerase; SSV: somatic short variant; VAF: variant allele frequency; WGS: whole-genome sequencing.

Ethics Approval and Consent to Participate

All patients participating in the study gave their informed consent. The study and the use of all clinical materials have been approved by the Ethics Committee of the Hospital District of Southwest Finland under decision number VARHA/28314/13.02.02/2023.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

The authors acknowledge CSC-IT Center for Science, Finland, for computational resources. ChatGPT [87] and Grammarly [88] were used to improve the grammar, vocabulary, and the flow of the text written by the authors. Zenodo includes queries and output by Chat GPT3.5 and GPT4 to improve drafts of the manuscript [91]. K.L. acknowledges the Biomedicum Helsinki Foundation for a personal research grant.

Author Contributions

Kari Lavikka (Conceptualization, Methodology, Software, Formal analysis, Investigation, Data curation [equal], Writing—original draft, Writing—review & editing, Visualization), Jaana Oikkonen (Conceptualization [supporting], Data curation [equal], Formal analysis [equal], Investigation [supporting], Writing—review & editing [supporting]), Yilin Li (Data curation [equal], Formal analysis [equal], Investigation [supporting], Writing—review & editing [supporting]), Taru Muranen (Data curation [equal], Formal analysis [equal], Investigation [supporting], Writing—review & editing [supporting]), Giulia Micoli (Formal analysis [equal]), Giovanni Marchi (Formal analysis [equal]), Alexandra Lahtinen (Investigation [supporting], Writing—review & editing [supporting]), Kaisa Huhtinen (Data curation [equal], Resources [equal]), Rainer Lehtonen (Conceptualization [supporting], Writing—review & editing [supporting]), Sakari Hietanen (Resources [equal], Writing—review & editing [supporting]), Johanna Hynninen (Resources [equal], Writing—review & editing [supporting]), Anni Virtanen (Data curation [equal], Investigation [supporting], Writing—original draft [supporting], Writing—review & editing [supporting]), and Sampsa Hautaniemi (Funding acquisition [lead], Project administration [lead], Resources [equal], Writing—review & editing [supporting])

Funding

This project received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 965193 for DECIDER (S.Ha.) and No. 847912 for RESCUER (S.Ha.), the Academy of Finland (S. Ha., project No. 325956), Sigrid Jusélius Foundation (S.Ha.), and the Cancer Foundation Finland (S.Ha.). Open access funded by Helsinki University Library.

References

1. Nielsen CB, Cantor M, Dubchak I, et al. Visualizing genomes: techniques and challenges. Nat Methods 2010;7:S5–S15. https://doi.org/10.1038/nmeth.1422.
2. O'Donoghue SI, Baldi BF, Clark SJ, et al. Visualization of biomedical data. Annu Rev Biomed Data Sci 2018;1:275–304. https://doi.org/10.1146/annurev-biodatasci-080917-013424.
3. Nusrat S, Harbig T, Gehlenborg N. Tasks, techniques, and tools for genomic data visualization. Comput Graphics Forum 2019;38:781–805. https://doi.org/10.1111/cgf.13727.
4. Diesh C. Awesome genome visualization. https://cmdcolin.github.io/awesome-genome-visualization/. Accessed 2024 July 2.
5. Van Den Brandt A, L'Yi S, Nguyen HN, et al. Understanding visualization authoring techniques for genomics data in the context of personas and tasks. OSF Preprints 2024. https://doi.org/10.17605/OSF.IO/BDJ4V. Accessed 2024 July 2.
6. Bostock M, Ogievetsky V, Heer J. D^3 data-driven documents. IEEE Trans Visual Comput Graphics 2011;17:2301–9. https://doi.org/10.1109/TVCG.2011.185.
7. Diesh C, Stevens GJ, Xie P, et al. JBrowse 2: a modular genome browser with views of synteny and structural variation. Genome Biol 2023;24. https://doi.org/10.1186/s13059-023-02914-z.
8. Wickham H. A layered grammar of graphics. J Comput Graph Statist 2010;19:3–28. https://doi.org/10.1198/jcgs.2009.07098.
9. Satyanarayan A, Moritz D, Wongsuphasawat K, et al. Vega-Lite: a grammar of interactive graphics. IEEE Trans Visual Comput Graphics 2017;23:341–50. https://doi.org/10.1109/TVCG.2016.2599030.
10. L'Yi S, Wang Q, Lekschas F, et al. Gosling: a grammar-based toolkit for scalable and interactive genomics data visualization. IEEE Trans Visual Comput Graphics 2022;28:140–50. https://doi.org/10.1109/TVCG.2021.3114876.
11. Yin T, Cook D, Lawrence M. ggbio: an R package for extending the grammar of graphics for genomic data. Genome Biol 2012;13:R77. https://doi.org/10.1186/gb-2012-13-8-r77.
12. Wilkinson L. The Grammar of Graphics. 2nd ed. 2005. New York, NY: Springer-Verlag;
13. Thorvaldsdóttir H, Robinson JT, Mesirov JP. Integrative Genomics Viewer (IGV): high-performance genomics data visualization and

exploration. Briefings Bioinf 2013;14:178–92. https://doi.org/10.1093/bib/bbs017.
14. Robinson JT, Thorvaldsdottir H, Turner D, et al. igv.Js: an embeddable JavaScript implementation of the Integrative Genomics Viewer (IGV). Bioinformatics 2023;39. https://doi.org/10.1093/bioinformatics/btac830.
15. Lee CM, Barber GP, Casper J, et al. UCSC Genome Browser enters 20th year. Nucleic Acids Res 2019;48:D756–D761. https://doi.org/10.1093/nar/gkz1012.
16. Elmqvist N, Moere AV, Jetter H-C, et al. Fluid interaction for information visualization. Information Visualization 2011;10:327–40. https://doi.org/10.1177/1473871611413180.
17. Lavikka K. GenomeSpy website. https://genomespy.app/. Accessed 2024 January 4.
18. Lavikka K, Oikkonen J, Li Y, et al. GenomeSpy visualization: DECIDER clinical trial. 2024. https://csbi.ltdk.helsinki.fi/p/genomespy-paper-2024/. Accessed 2024 May 6.
19. Gadducci A, Guarneri V, Peccatori FA, et al. Current strategies for the targeted treatment of high-grade serous epithelial ovarian cancer and relevance of BRCA mutational status. J Ovarian Res 2019;12. https://doi.org/10.1186/s13048-019-0484-6.
20. Torre LA, Trabert B, DeSantis CE, et al. Ovarian cancer statistics, 2018. CA A Cancer J Clinicians 2018;68:284–96. https://doi.org/10.3322/caac.21456.
21. Macintyre G, Goranova TE, De Silva D, et al. Copy number signatures and mutational processes in ovarian carcinoma. Nat Genet 2018;50:1262–70. https://doi.org/10.1038/s41588-018-0179-8.
22. Bell D, Berchuck A, Birrer M, et al. Integrated genomic analyses of ovarian carcinoma. Nature 2011;474:609–15. https://doi.org/10.1038/nature10166.
23. Kasherman L, Garg S, Tchrakian N, et al. Can TP53 variant negative be high-grade serous ovarian carcinoma? A case series. Gynecol Oncol Rep 2021;36:100729. https://doi.org/10.1016/j.gore.2021.100729.
24. Zarei S, Wang Y, Jenkins SM, et al. Clinicopathologic, immunohistochemical, and molecular characteristics of ovarian serous carcinoma with mixed morphologic features of high-grade and low-grade serous carcinoma. Am J Surg Pathol 2020;44:316–28. https://doi.org/10.1097/PAS.0000000000001419.
25. Lavikka K. GenomeSpy Observable notebooks. https://observablehq.com/collection/@tuner/genomespy. Accessed 2024 January 4.
26. DePristo MA, Banks E, Poplin R, et al. A framework for variation discovery and genotyping using next-generation DNA sequencing data. Nat Genet 2011;43:491–98. https://doi.org/10.1038/ng.806.
27. Liu Z, Heer J. The effects of interactive latency on exploratory visual analysis. IEEE Trans Visual Comput Graphics 2014;20:2122–31. https://doi.org/10.1109/TVCG.2014.2346452.
28. Heer J, Robertson GG. Animated transitions in statistical data graphics. IEEE Trans Visual Comput Graphics 2007;13:1240–47. https://doi.org/10.1109/TVCG.2007.70539.
29. SegmentModel Spy. https://genomespy.app/segmentmodel/. Accessed 2024 January 4.
30. Ragan ED, Endert A, Sanyal J, et al. Characterizing provenance in visualization and data analysis: an organizational framework of provenance types and purposes. IEEE Trans Visual Comput Graphics 2016;22:31–40. https://doi.org/10.1109/TVCG.2015.2467551.
31. Gratzl S, Lex A, Gehlenborg N, et al. From visual exploration to storytelling and back again. Comput Graphics Forum 2016;35:491–500. https://doi.org/10.1111/cgf.12925.
32. GenomeSpy Playground. https://genomespy.app/playground/. Accessed 2024 June 16.
33. Amemiya HM, Kundaje A, Boyle AP. The ENCODE blacklist: identification of problematic regions of the genome. Sci Rep 2019;9. https://doi.org/10.1038/s41598-019-45839-z.
34. O'Leary NA, Wright MW, Brister JR, et al. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. Nucleic Acids Res 2016;44:D733–45. https://doi.org/10.1093/nar/gkv1189.
35. Huang D, Savage SR, Calinawan AP, et al. A highly annotated database of genes associated with platinum resistance in cancer. Oncogene 2021;40:6395–405. https://doi.org/10.1038/s41388-021-02055-2.
36. Lavikka K. DECIDER visualization specification. Zenodo. 2024. https://doi.org/10.5281/zenodo.11121377.
37. Baslan T, Morris JP, Zhao Z, et al. Ordered and deterministic cancer genome evolution after p53 loss. Nature 2022;608:795–802. https://doi.org/10.1038/s41586-022-05082-5.
38. Tavassoli M, Ruhrberg C, Beaumont V, et al. Whole chromosome 17 loss in ovarian cancer. Genes Chromosomes Cancer 1993;8:195–98. https://doi.org/10.1002/gcc.2870080310.
39. GenomeSpy bookmark: TP53 and LOH in chr17 https://csbi.ltdk.helsinki.fi/p/genomespy-paper-2024/#bookmark:TP53-and-LOH-in-chr17. Accessed 2024 June 16.
40. Cerretelli G, Ager A, Arends MJ, et al. Molecular pathology of Lynch syndrome. J Pathol 2020;250:518–31. https://doi.org/10.1002/path.5422.
41. Shneiderman B. Direct manipulation: a step beyond programming languages. Computer 1983;16:57–69. https://doi.org/10.1109/MC.1983.1654471.
42. GenomeSpy bookmark: high and low number of breakpoints. https://csbi.ltdk.helsinki.fi/p/genomespy-paper-2024/#bookmark:High-and-low-number-of-breakpoints. Accessed 2024 Jun 16.
43. Popova T, Manié E, Boeva V, et al. Ovarian cancers harboring inactivating mutations in CDK12 display a distinct genomic instability pattern characterized by large tandem duplications. Cancer Res 2016;76:1882–91. https://doi.org/10.1158/0008-5472.CAN-15-2128.
44. GenomeSpy bookmark: Top 5 fragmented patients. https://csbi.ltdk.helsinki.fi/p/genomespy-paper-2024/#bookmark:Top-5-fragmented-patients. Accessed 2024 June 16.
45. Slomovitz B, Gourley C, Carey MS, et al. Low-grade serous ovarian cancer: state of the science. Gynecol Oncol 2020;156:715–25. https://doi.org/10.1016/j.ygyno.2019.12.033.
46. Hunter SM, Anglesio MS, Ryland GL, et al. Molecular profiling of low grade serous ovarian tumours identifies novel candidate driver genes. Oncotarget 2015;6:37663–77. https://doi.org/10.18632/ONCOTARGET.5438.
47. Murali R, Selenica P, Brown DN, et al. Somatic genetic alterations in synchronous and metachronous low-grade serous tumours and high-grade carcinomas of the adnexa. Histopathology 2019;74:638–50. https://doi.org/10.1111/HIS.13796.
48. Kircher M, Witten DM, Jain P, et al. A general framework for estimating the relative pathogenicity of human genetic variants. Nat Genet 2014;46:310–15. https://doi.org/10.1038/ng.2892.
49. Beroukhim R, Getz G, Nghiemphu L, et al. Assessing the significance of chromosomal aberrations in cancer: methodology and application to glioma. Proc Natl Acad Sci USA 2007;104:20007–12. https://doi.org/10.1073/pnas.0710052104.

50. GenomeSpy bookmark: copy-number landscape. https://csbi.ltdk.helsinki.fi/p/genomespy-paper-2024/#bookmark:Copy-number-landscape. Accessed 2024 June 16.
51. Etemadmoghadam D, Au-Yeung G, Wall M, et al. Resistance to CDK2 inhibitors is associated with selection of polyploid cells in CCNE1-amplified ovarian cancer. Clin Cancer Res 2013;19:5960–71. https://doi.org/10.1158/1078-0432.CCR-13-1337.
52. GenomeSpy bookmark: WGD and CCNE1. https://csbi.ltdk.helsinki.fi/p/genomespy-paper-2024/#bookmark:WGD-and-CCNE1. Accessed 2024 June 16.
53. Cameron DL, Baber J, Shale C, et al. GRIDSS2: comprehensive characterisation of somatic structural variation using single breakend variants and structural variant phasing. Genome Biol 2021;22. https://doi.org/10.1186/s13059-021-02423-x.
54. Landrum MJ, Lee JM, Benson M, et al. ClinVar: improving access to variant interpretations and supporting evidence. Nucleic Acids Res 2018;46:D1062–67. https://doi.org/10.1093/nar/gkx1153.
55. GenomeSpy visualization: experimental GRIDSS pipeline. https://csbi.ltdk.helsinki.fi/p/genomespy-paper-2024/GRIDSS/. Accessed 2024 June 16.
56. Schroeder MP, Gonzalez-Perez A, Lopez-Bigas N. Visualizing multidimensional cancer genomics data. Genome Med 2013;5:9. https://doi.org/10.1186/gm413.
57. Liu Y, Chen C, Xu Z, et al. Deletions linked to TP53 loss drive cancer through p53-independent mechanisms. Nature 2016;531:471–75. https://doi.org/10.1038/nature17157.
58. Satyanarayan A, Lee B, Ren D, et al. Critical reflections on visualization authoring systems. IEEE Trans Visual Comput Graphics 2019;26:461–471. https://doi.org/10.1109/TVCG.2019.2934281.
59. Satyanarayan A, Heer J. Lyra: an interactive visualization design environment. Comput Graphics Forum 2014;33:351–60. https://doi.org/10.1111/cgf.12391.
60. Pandey A, L'Yi S, Wang Q, et al. GenoREC: a recommendation system for interactive genomics data visualization. IEEE Trans Vis Comput Graphics 2023;29:570–80. https://doi.org/10.1109/TVCG.2022.3209407.
61. VanderPlas J, Granger B, Heer J, et al. Altair: interactive statistical visualizations for Python. J Open Source Softw 2018;3:1057. https://doi.org/10.21105/joss.01057.
62. Tavares G. TWGL: a tiny WebGL helper library. https://twgljs.org/. Accessed 2024 January 4.
63. Satyanarayan A, Russell R, Hoffswell J, et al. Reactive Vega: a streaming dataflow architecture for declarative interactive visualization. IEEE Trans Visual Comput Graphics 2016;22:659–68. https://doi.org/10.1109/TVCG.2015.2467091.
64. Mark Erikson: Redux Toolkit. https://redux-toolkit.js.org/. Accessed 2024 January 4.
65. Google LLC: Lit. Google LLC. https://lit.dev/. Accessed 2024 January 4.
66. Bolger AM, Lohse M, Usadel B. Trimmomatic: a flexible trimmer for Illumina sequence data. Bioinformatics 2014;30:2114–20. https://doi.org/10.1093/bioinformatics/btu170.
67. Andrews S. FastQC. https://www.bioinformatics.babraham.ac.uk/projects/fastqc/. Accessed 2024 January 4.
68. Cervera A, Rantanen V, Ovaska K, et al. Anduril 2: upgraded large-scale data integration framework. Bioinformatics 2019;35:3815–17. https://doi.org/10.1093/bioinformatics/btz133.
69. Broad Institute. Picard toolkit. https://broadinstitute.github.io/picard/. Accessed 2024 January 4.
70. Li H. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. arXiv 2013. https://doi.org/10.48550/arXiv.1303.3997. Accessed 2024 July 2.
71. McKenna A, Hanna M, Banks E, et al. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. Genome Res 2010;20:1297–303. https://doi.org/10.1101/gr.107524.110.
72. Benjamin D, Sato T, Cibulskis K, et al. Calling somatic SNVs and indels with Mutect2. Biorxiv. 2019. https://doi.org/10.1101/861054. Accessed 2024 July 2.
73. Poplin R, Ruano-Rubio V, DePristo MA, et al. Scaling accurate genetic variant discovery to tens of thousands of samples. Biorxiv. 2018. https://doi.org/10.1101/201178. Accessed 2024 July 2.
74. Wang K, Li M, Hakonarson H. ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. Nucleic Acids Res 2010;38:e164. https://doi.org/10.1093/nar/gkq603.
75. Alexandrov LB, Kim J, Haradhvala NJ, et al. The repertoire of mutational signatures in human cancer. Nature 2020;578:94–101. https://doi.org/10.1038/s41586-020-1943-3.
76. Lavikka K. ASCAT Algorithm for GATK Segments (Version 1.0) [Computer software]. Software Heritage. https://archive.softwareheritage.org/swh:1:snp:6c4c501da1206a6b5bc3f97e7e9dc543d26a11dd;origin=https://github.com/tuner/ASCAT-for-GATK. Accessed 2024 June 18.
77. Van Loo P, Nordgard SH, Lingjaerde OC, et al. Allele-specific copy number analysis of tumors. Proc Natl Acad Sci USA 2010;107:16910–15. https://doi.org/10.1073/pnas.1009843107.
78. PureCN GitHub issue: copy ratio adjustments for purity/ploidy are incorrect. https://github.com/lima1/PureCN/issues/40. Accessed 2024 June 18.
79. Hartwig Medical Foundation. hmftools: GRIPSS. https://github.com/hartwigmedical/hmftools/tree/master/gripss. Accessed 2024 January 5.
80. Priestley P, Baber J, Lolkema MP, et al. Pan-cancer whole-genome analyses of metastatic solid tumours. Nature 2019;575:210–16. https://doi.org/10.1038/s41586-019-1689-y.
81. Hartwig Medical Foundation. hmftools: AMBER. https://github.com/hartwigmedical/hmftools/tree/master/amber. Accessed 2024 January 5.
82. Hartwig Medical Foundation. hmftools: COBALT. https://github.com/hartwigmedical/hmftools/tree/master/cobalt. Accessed 2024 January 5.
83. Mermel CH, Schumacher SE, Hill B, et al. GISTIC2.0 facilitates sensitive and confident localization of the targets of focal somatic copy-number alteration in human cancers. Genome Biol 2011;12. https://doi.org/10.1186/gb-2011-12-4-r41.
84. Dolgin E. The most popular genes in the human genome. Nature 2017;551:427–31. https://doi.org/10.1038/d41586-017-07291-9.
85. Kerpedjiev P, Abdennur N, Lekschas F, et al. HiGlass: web-based visual exploration and analysis of genome interaction maps. Genome Biol 2018;19. https://doi.org/10.1186/s13059-018-1486-1.
86. GenomeSpy GitHub repository. https://github.com/genome-spy/genome-spy/. Accessed 2024 June 17.
87. OpenAI: ChatGPT (GPT-3.5 and GPT-4) [Large language model]. 2023. https://chat.openai.com/chat. Accessed 2024 May 10.
88. Grammarly, Inc. Grammarly. https://www.grammarly.com/. Accessed 2024 May 10.
89. Lavikka K. GenomeSpy—A Visualization Grammar and GPU-accelerated Toolkit for Genomic Data (Version 0.53.1) [Computer software]. Software Heritage. https://archive.softwareheritage.org/swh:1:snp:f87cb6eb27ddbabecb203b887ed2ded75c4d59aa;origin=https://github.com/genome-spy/genome-spy. Accessed 2024 July 2.

90. Lavikka K. SegmentModel Spy—An Interactive Visualization Tool for GATK CNV Analysis (Version 1.0) [Computer software]. Software Heritage. https://archive.softwareheritage.org/swh:1:snp:0333e6dce48d517d08900025b6efc48f2a15b7ee;origin=https://github.com/genome-spy/segment-model-spy. Accessed 2024 July 2.

91 Lavikka K.(Transcripts Demonstrating the Application of ChatGPT in the Composition of the Manuscript "Deciphering Cancer Genomes with GenomeSpy: A Grammar-Based Visualization Toolkit" by Lavikka, et al. Zenodo [Dataset] 2024). https://doi.org/10.5281/zenodo.12775114.

Received: January 9, 2024. **Revised:** May 13, 2024. **Accepted:** June 19, 2024