Supplementary Information for

Oviz-Bio: a web-based platform for interactive cancer genomics data visualization

Wenlong Jia, Hechen Li, Shiying Li, Lingxi Chen, and Shuai Cheng Li*

* **Corresponding author:**
Shuai Cheng Li: shuaicli@cityu.edu.hk

This file includes:
Supplementary Text S1
Supplementary Figures S1 to S6

Supplementary Text S1: Supplementary information on input files for eleven cancer genomic data visualizations in Oviz-Bio

The documentation and demo data of all visualizations on Oviz-Bio are available in the GitHub repository (https://github.com/Nobel-Justin/Oviz-Bio-demo).

1 'Mut on Genes' visualization

The 'Mut on Genes' accepts two input files: one mutation file (required) and one depth TSV file (optional). Note that the depth TSV file should bgzip compressed (see below). We provide demo files in Oviz-Bio GitHub repository.

1.1 mutation file

The mutation file could be standard MAF file or VCF file. Or, a simple CSV file in format specified below.

#SampleID	alt_type	chr	pos	ref_allele	alt_allele	gene
TCGA-AK-3451	SNP	chr3	10183646	G	A	VHL
TCGA-BP-4798	INS	chr3	10183729	-	A	VHL
TCGA-BP-4961	DEL	chr3	10183714	CGTGCTG	-	VHL

- the first line should be the header as specified above.
- the prefix (#) is mandatory to indicate the header line.
- currently, alt_type allows SNP, DEL, and INS.

1.2 Depth TSV BGZ File (Optional)

User could upload a tsv.bgz compressed file to supply the depth distribution of the gene region. The uploaded file must match the required format as specified below.

#chr	pos	depth
chr3	10183646	90
chr3	...	...
chr3	10183710	95

- the first line should be the header as specified above.
- the prefix (#) is mandatory to indicate the header line.

The TSV file must be sorted by chromosome and position, and compressed by bgzip tools for tabix indexing to support fast data processing at the backend of Oviz-Bio. For example, run the following command in the linux terminal (bgzip installed):

$ (head -1 depth.tsv; sed -n '2,$p' depth.tsv | sort -k1,1 -k2n) | bgzip -c > depth.tsv.bgz

1.3 Backend annotation task

Note that backend annotation task will be activated once mutation csv file is uploaded. User could check the job status in the task monitor at the bottom right of analysis page. The annotation software is ANNOVAR, and the relevant options are: '-protocol avsnp150,ensGene,cosmic70 -operation f,g,f'

2 'SNV Context' visualization

The 'SNV Context' accepts two input files: one SNV file (required) and one region BED file (optional). We provide demo files in Oviz-Bio GitHub repository.

2.1 SNV file

This input file could be standard MAF file (see format) or VCF file. Or, a simple TSV file in the format specified below.

#contig	position	context	ref_allele	alt_allele	tumor_f
chr1	101686	AxA	A	G	0.113636

- the first line should be the header as specified above.
- the prefix (#) is mandatory to indicate the header line.
- 'contig' and 'position' respectively stand for the chromosome and position of the mutation.
- 'ref_allele' and 'alt_allele' respectively stand for the base before and after the mutation.
- users can filter mutations with 'tumor_f' value by custom condition in the sidebar.
- NOTE that the 'tumor_f' is optional, i.e., this column could be omitted.

2.2 Custom BED File (optional)

To calculate the mutation density, we provide the default region used in our website, which is the whole genome sequence. In addition, a custom BED file allows user to replace it. We will filter out mutations that are not in the custom region during the calculation. Note that the uploaded BED file is only applied when you choose the custom bed option in the Settings section of the sidebar.

The uploaded BED file must match the required format as specified below (please keep the header).

#chr	startPos	endPos
chr1	11174854	11175054
chr1	11183826	11185871

3 'SNV Signature' visualization

The 'SNV Signature' requires one Signature CSV input file. We provide demo files in Oviz-Bio GitHub repository.

Signature Data (CSV file)

The uploaded Signature CSV file must have the header in the required format as specified below.

types	subtypes	signature 1	signature 2	signature 3	signature 4	signature 5
C>A	ACA	0.000353018	0.000284683	0.000401966	0.002938016	0.002984077

- the first line should be the header as specified above.
- The 'types' and 'subtypes' stand for the six substitution subtypes and the adjacent bases, respectively. The values of these two columns should also follows the format shown above.
- The 'signature 1' to 'signature 5' each stands for a signature. The name, such as 'signature 1', is not fixed and can be replaced by any other text.
- The number of signatures is not limited.
- Note that this file must contain all 96 mutation types, i.e., 96 rows.

4 'Signature Dist' visualization

The 'Signature Dist' requires one Signature Dist CSV input file. We provide demo files in Oviz-Bio GitHub repository.

Signature Dist Data (CSV file)

The uploaded Signature Dist CSV file must have the header in the required format as specified below. The input is mutation signature compositions in batch of samples using NMF algorithm, e.g., decipherMutationalSignatures.

Observations	T01	T02	T03	T04	T05	T06	T07
signature 1	5.364	21.102	13.466	3.781	17.510	10.632	25.913
signature 2	...	...	...	...	...	...	...
signature ..	...	...	...	...	...	...	...
signature N	...	...	...	...	...	...	...

- the first line should be the header as specified above.
- 'Observations' is the signature name.
- Keys like 'T01' is the sample ID.
- The number of sample is not limited.
- Each row is the measure of a specified signature in each individual sample. Note that this measure does not have to be normalized to 100%.

5 'CNV Haplotype View' visualization

The 'CNV Haplotype View' requires user to upload the CSV output from the 'Patchwork'. We provide demo files in Oviz-Bio GitHub repository.

	chr	start	end	snvs	ai	median	Cn	mCn	fullCN
1	chr1	19801	700386	63	0.360098826	0.961870155	2	1	cn2m1

6 'CNV Focal Cluster' visualization

The 'CNV Focal Cluster' requires three input files: the GISTIC scores TSV file, amp_genes TSV file, and del_genes TSV file, all of which are results from GISTIC. We provide demo files in Oviz-Bio GitHub repository.

6.1 GISTIC scores TSV file

The GISTIC scores TSV file has the header in the format as specified below.

Type	chromsome	Start	End	q-value	G-score	average amplitude	frequency
Amp	1	7500	167500	0.133141	0.208099	0.413824	0.333333
...	...	...	...	...	...	...	...
Del	7	94927500	95392500	0	0.072177	0.231943	0.25

6.2 amp_genes TSV file

The amp genes file contains amplification peaks identified in the GISTIC analysis. The first four rows are 'cytoband', 'q-value', 'residual q-value' and 'wide peak boundaries'. The remaining rows list the genes contained in each wide peak. For peaks that contain no genes, the nearest gene is listed in brackets.

6.3 del_genes TSV file

The del genes file contains one column for each deletion peak identified in the GISTIC analysis. The file format for the del genes file is identical to the format for the amp genes file.

7 'SV Reads Support' visualization

The 'SV Reads Support' requires one input files: Read Support TXT file. We provide demo files in Oviz-Bio GitHub repository.

Read Support TXT file

User can generate the input files using read_support.py with BAM file and SvABA generated sv VCF file. Please check https://github.com/paprikachan/ComplexSV for source code and usage of read_support.py.

The input file stores split-reads for given SVs. Each SV event starts with 'sv' section, and then split-reads details.

The header description of the 'sv' section is shown below.

header	example	description
chrom_5p	chr11	chromosome of 5' breakpoint
bkpos_5p	100000	position of 5' breakpoint
strand_5p	+	strand of 5' breakpoint
chrom_3p	chr1	chromosome of 3' breakpoint
bkpos_3p	101000	position of 3' breakpoint
strand_3p	-	strand of 3' breakpoint
innser_ins	TACCGATAT	inner insertion at breakpoint, or NONE
sv_meta_info	HM:TCA,-3	micro-homology (TCA) -3 index
splits_read_num	10	the number of supporting split read pairs

The split-reads details section contains six columns.

column	description
read_query_name	the read id
read_flags	reads status (see below)
read_position	reads mapped position (see below)
read_seq	reads sequence
read_qual	reads base quality
read_meta_info	meta info (see below)

The 'read_flags' consists of five flag j, J, r, R, d.

Flag	Description
j	The current read is split read
J	The paired read of current read is split read
r	The current read is reverse read
R	The paired read of current read is reverse read
d	The current read is duplicated

The 'read_position' is recorded by the related position aligned to reconstructed SV haplotype. See instance below.

```
   |<-    5'segment    ->| inner_ins |<- 3'  segment   ->|
   |xxxxxxxxxxxxxxxxTCA|iiiiiiiiiii|xxxxxxxxxxxxxxxxxx|
                      :    ^
          negative    : <- 0 -> positive
                      :
   split_read:        xxxxxxxxxxxxxxxxxxxxxxxxxxxx
                      ^
   read_position: -5

   micro-homology:      TCA
                        ^
   HM position         -2
```

The 'read_meta_info', stores reads information, e.g. 'BX:NONE;INS:-12(3),15(2);DEL:-86(1),36(4)'.

- 'BX:GACACTAGTTAAGATG-1' is the barcode of the reads, or else NONE.
- 'INS:-12(3),15(2)' shows the small insertions on reads. '-12(3)' means that the first base of the insertion has 3 base pairs and its index is -12, and '15(2)' means that the first base of the second insertion has 2 base pairs and locates at index 15. Note that the insertion sequence will not be displayed on reads, but marked with a yellow inverted triangle at the insertion position.
- 'DEL:-86(1),36(4)' shows the small deletaion on reads. '36(4)' means that the front base of the deleted fragment is index 36, and the length of the deleted fragment is 4 bases. The deletion sequence is filled with red short line on reads.

8 'SV Heatmap' visualization

The 'SV Heatmap' requires one input files: Heatmap Data TXT file. We provide demo files in Oviz-Bio GitHub repository.

Heatmap Data TXT file

User can generate the input files using linkage_heatmap.py with BAM file and SvABA generated sv VCF file. Please check https://github.com/paprikachan/ComplexSV for source code and usage of linkage_heatmap.py.

The input file starts with 'sv' section. The header description of the 'sv' section is shown below.

```
#sv
chr11    76139879    +    chr2    63328429    -    VARTYPE=BND:TRX-tt
```

chrom_5p	bkpos_5p	strand_5p	chrom_3p	bkpos_3p	strand_3p	meta_info
chr11	76139879	+	chr2	63328429	-	VARTYPE=BND:TRX-tt

- The 'chrom_5p', 'bkpos_5p', 'strand_5p' respectively stands for the chromosome, position, strand of 5' breakpoint.
- The 'chrom_3p', 'bkpos_3p', 'strand_3p' respectively stands for the chromosome, position, strand of 3' breakpoint.
- The 'meta_info' stores the meta information, such as the variation type of the SV.

Next is the 'heatmap' section, it starts with the header:

```
#heatmap    linkage_type=10x barcode
```

- The 'linkage_type' specifies the type of read linkage used to count, now we support to show '10x barcode' and '10x barcode' linkage type.

From the two breakpoints of one given SV event, expanding the breakpoint with 1000bp, we can generate four region pairs and their shared linkage count matrix.

Region pair 1: vertical region is (bkpos_5p-1000bp, bkpos_5p+1000bp) and horizontal region is (bkpos_5p-1000bp, bkpos_5p+1000bp). This is the second quadrant of heatmap.

```
v=chr2:63327429-63329429    h=chr2:63327429-63329429    resolution=100    axis=left-top
63.0,34.0,35.0,32.0,21.0,...
34.0,82.0,56.0,37.0,36.0,...
35.0,56.0,104.0,47.0,40.0,...
32.0,37.0,47.0,84.0,40.0,...
21.0,36.0,40.0,40.0,83.0,...
...
```

Region pair 2: vertical region is (bkpos_5p-1000bp, bkpos_5p+1000bp) and horizontal region is (bkpos_3p-1000bp, bkpos_3p+1000bp). This is the first quadrant of heatmap.

```
v=chr2:63327429-63329429    h=chr11:76138879-76140879    resolution=100    axis=right-top
8.0,14.0,7.0,8.0,11.0,...
9.0,17.0,13.0,14.0,13.0,...
14.0,16.0,17.0,22.0,15.0,...
9.0,14.0,11.0,13.0,13.0,...
8.0,13.0,13.0,16.0,13.0,...
...
```

Region pair 3: vertical region is (bkpos_3p-1000bp, bkpos_3p+1000bp) and horizontal region is (bkpos_5p-1000bp, bkpos_5p+1000bp). This is the third quadrant of heatmap.

```
v=chr11:76138879-76140879    h=chr2:63327429-63329429    resolution=100    axis=left-bottom
8.0,9.0,14.0,9.0,8.0,...
14.0,17.0,16.0,14.0,13.0,...
7.0,13.0,17.0,11.0,13.0,...
8.0,14.0,22.0,13.0,16.0,...
11.0,13.0,15.0,13.0,13.0,...
...
```

Region pair 4: vertical region is (bkpos_3p-1000bp, bkpos_3p+1000bp) and horizontal region is (bkpos_3p-1000bp, bkpos_3p+1000bp). This is the fourth quadrant of heatmap.

```
v=chr11:76138879-76140879    h=chr11:76138879-76140879    resolution=100    axis=right-bottom
80.0,52.0,27.0,42.0,33.0,...
52.0,140.0,56.0,61.0,50.0,...
27.0,56.0,124.0,72.0,50.0,...
42.0,61.0,72.0,152.0,57.0,...
33.0,50.0,50.0,57.0,112.0,...
...
```

9 'Fusion Trans Junction' visualization

The 'Fusion Trans Junction' requires one Fusion TSV input file. We provide demo files in Oviz-Bio GitHub repository.

Fusion TSV file

#Sample-ID	up_gene	up_chr	up_pos	up_covlen	down_gene	down_chr	down_pos	down_covlen
T003	FGFR3	chr4	1808661	886	TACC3	chr4	1739325	704

- the first line should be the header as specified above.
- This file is easy to prepare from the result of common fusion tools, such as SOAPfuse.
- The 'up_pos' and 'down_pos' are genomic coordinates in the genome, currently we require the position must locate in exon region of relevant genes.
- The 'up_covlen' and 'down_covlen' respectively stands for the distance fusion supporting reads extend from the junction position of upstream (5') and downstream (3') fusion partern genes.
- NOTE that the 'up_covlen' and 'down_covlen' are optional, i.e., these two columns could be omitted.

10 'Virus Integ HotSpot' visualization

The 'Virus Integ HotSpot' requires one Virus Integration CSV input file. We provide demo files in Oviz-Bio GitHub repository.

Virus Integration CSV file

#SampleID	Chr	Position	Strand	JR_count
13T	chr5	1285509	-	5
13T	chr5	1285517	-	4
63T	chr5	1296259	-	16
...	...	...	...	...
34T	chr5	1297651	-	17
34T	chr5	1297639	-	2

- the first line should be the header as specified above.
- this file is easy to prepare from the result of common virus integration detection tools.
- The 'Strand' represents the relation DNA strand of virus integration. If the virus segment connects with host segment with different strands, the 'Strand' value should be '-', e.g., virus is plus strand and host is minus strand. If they connect in same strand, the 'Strand' value should be '+'.
- The 'JR_count' is the split-reads count of relevant virus integration. We consider 'JR_count' as the credibility of the virus integration. It will be used when page tries to display only one intergration for each sample, i.e., the sidebar option 'Unify samples' is enabled. The integration with the largest 'JR_count' will be selected.
- Note that the additional 'Gene' and 'Comments' columns in demo files are just notes, and this visualization will not load data of these two columns.

11 'LandScape' visualization

The 'LandScape' requires one LandScape Data CSV input file. We provide demo files in Oviz-Bio GitHub repository.

LandScape Data (CSV file)

The uploaded LandScape Data CSV file must match the required format as specified below. Here is an instance.

SampleID	g_TP53	g_PTEN	ht1_Gain	ht1_Loss	Individual-Info_gender	Clinical-Data_subtype
T001	Splice_Site	Missense	17	6	F	A

header
The first line of the file should be a header that contains column names as keys.

rows
Each row in the file should contain data for a sample.

samples ('SampleID' column)
The first column lists sample names in each row, with the key 'SampleID' in the header line. (mandatory)

gene-panel ('g_' prefix column)
Add columns for genes that you want to display. (mandatory)

- The header line keyword follows g_[GeneName] format, such as 'g_TP53' and 'g_PTEN'.
- Typically, the content of each table cell for one gene and one sample lists the mutations types, such as 'Missense', 'Synonymous', and 'Gain'. Use '-' for non-mutation, and 'N/A' for not-available.
- A cell may contain multiple mutation types (or values) using semicolon as separator, e.g., 'Mis;Loss'.
- Each cell allows three unique values at maximum. For example, 'Mis;Splice_Site;Mis;Loss' is OK, while 'Mis;Splice_Site;InDel;Mis;Gain' is not allowed because it contains 4 distinctive mutation types.
- All values appeared in 'g_' columns are summarized to calculate the sample frequency in each gene, which is shown by the horizontal histogram at the 'left-area' of gene-panel.
- Generally, in cancer research, genes are always annotated with more information, such as their pathway, GO ontology, certain comments, and P-value or Q-value representing their mutated

significance in batch samples. These information (if given) will be shown at the 'right-area' of gene-panel as tags, matrix or histogram. Please check details in online Manual of LandScape.

histogram ('ht' prefix column)
It is possible to add multiple stacked histograms that show attributes' distribution at the top of the visualization.

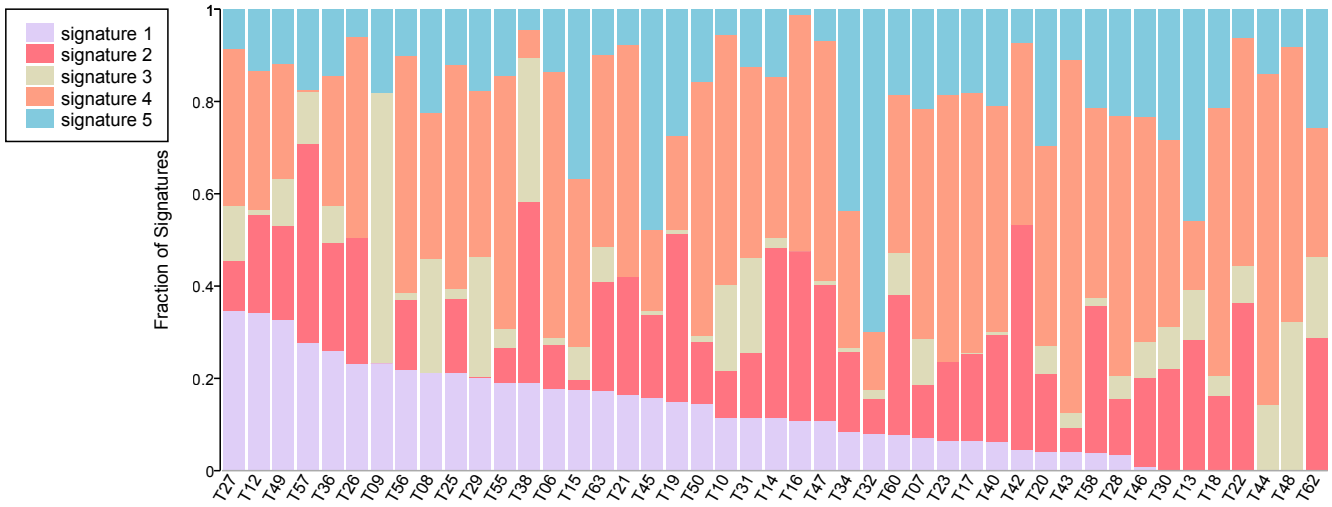
- Column names for histograms should follow the 'ht_[AttributeName]' format, such as 'ht_Missence', 'ht_Truncated', and 'ht_Gain'.
- Only numeric values (e.g., 15, 23.9 or 0.327) are allowed as cell content, and it keeps up to three decimals.
- Since the whole CSV file as a complete matrix, please use '-' to fill the intersecting cells between 'histogram' columns and 'gene-information' rows (i.e., these mentioned above: pathway, GO, comments, P-value, and Q-value, if exist).
- To display multiple histograms, use 'ht2_[attr]' for the second one, 'ht3_[attr]' for the third one, and so on. To allow flexibility, 'ht1_' and 'ht_' are both treated as the first one.
- Basically, histogram is simply named with their NO., e.g., 'ht2_' corresponds to 'Histogram 2'. Optionally, users can assign a name to each histogram by adding a bracketed value after 'ht'. For example, 'ht2(NewName)_' will be named as 'Histogram NewName' rather than 'Histogram 2'. The cooresponding section of this histogram in the sidebar will also use this new name.
- Although it's not common, user can omit 'ht' columns to avoid displaying any histogram in the figure.

additional panels
Add columns to show additional panels (more information on samples) at the bottom of the visualization. Each panel might contain several attributes belonging to one category, e.g., panel 'Individual-Info' contains attributes like age, gender, smoking.

- The column name key should follow the '[PanelName]_[AttributeName]' format, e.g., 'Individual-Info_age' and 'Individual-Info_gender'. Note that please avoid using '_' in PanelNames. Do not worry about panel naming because the sidebar provides options to customize the displaying name of each panel, where users can choose their preferred name such as 'Individual_Info' or 'Individual Info'.
- The cell content could be strings, numeric values, or 'N/A'. Please check details in online Manual of LandScape.
- We have two reserved PanelName short names: 'mtif' for 'Metadata' and 'pw' for 'Pathway'. Of course, you can just use the full original name as you wish.
- By default, all panels will be displayed from top to bottom on the page in the order in which they appear in the uploaded CSV file.

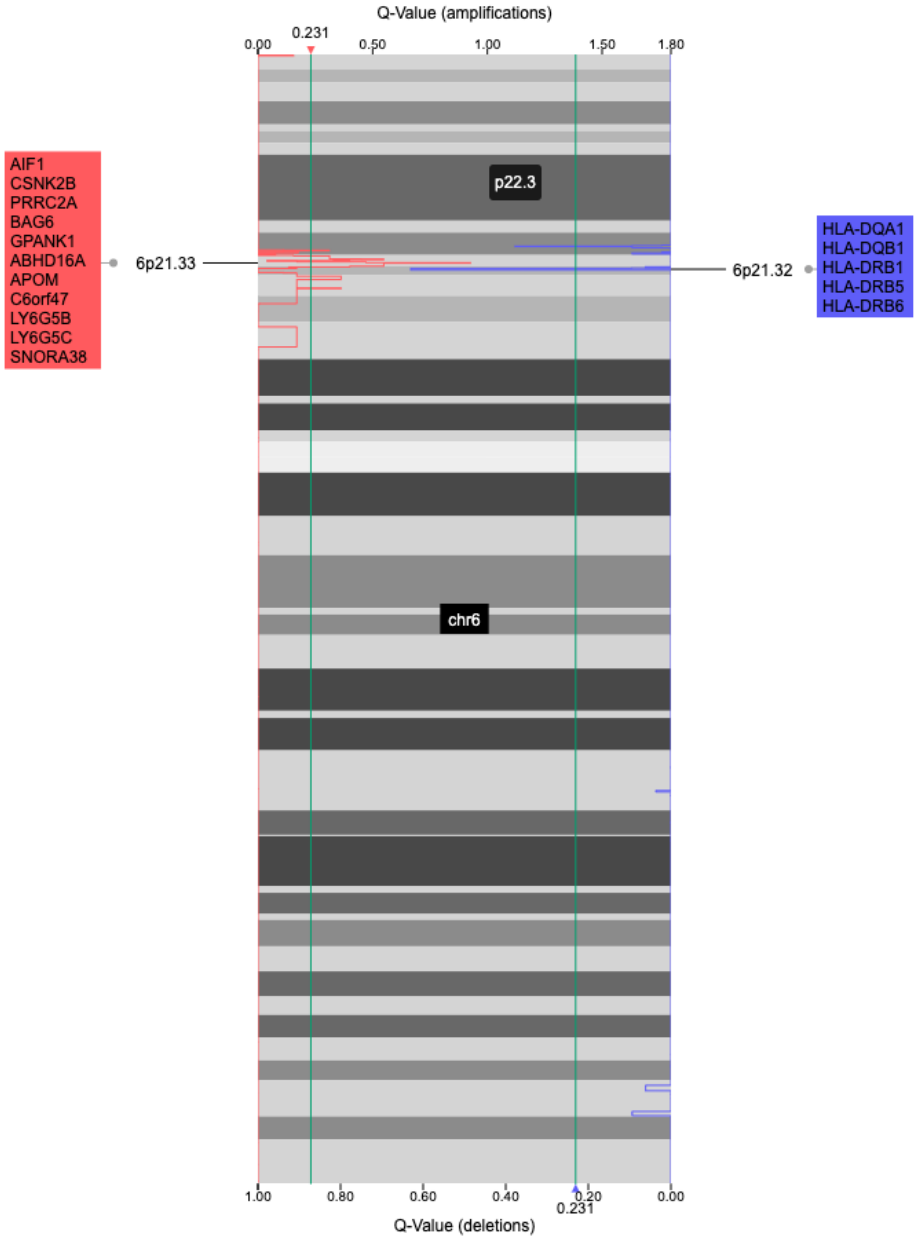
Supplementary Figures

Supplementary Figure S1



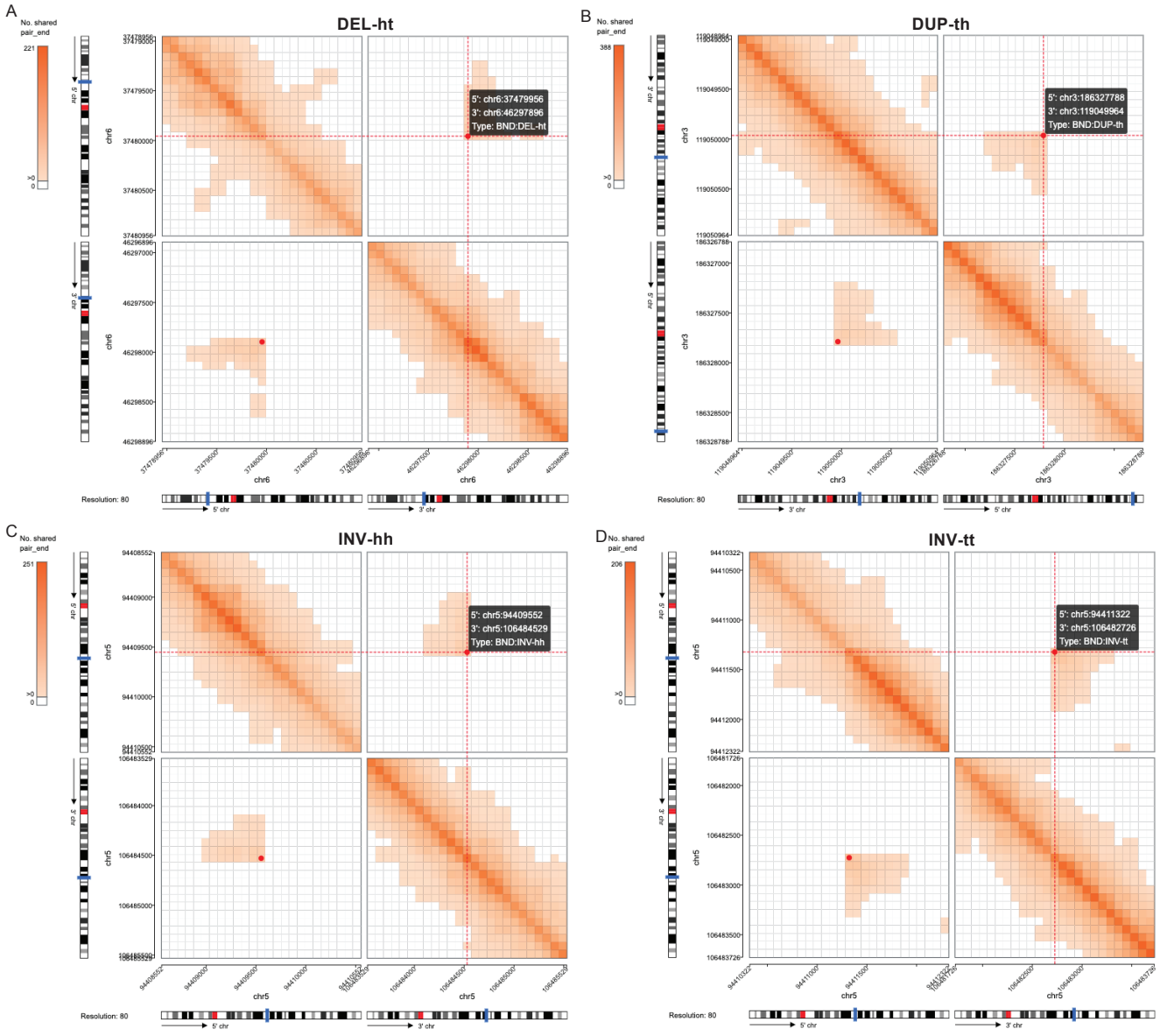
Supplementary Figure S1. Demo representation of the 'Signature Dist' visualization and features. Samples are sorted by fraction of the signature 1.

Supplementary Figure S2



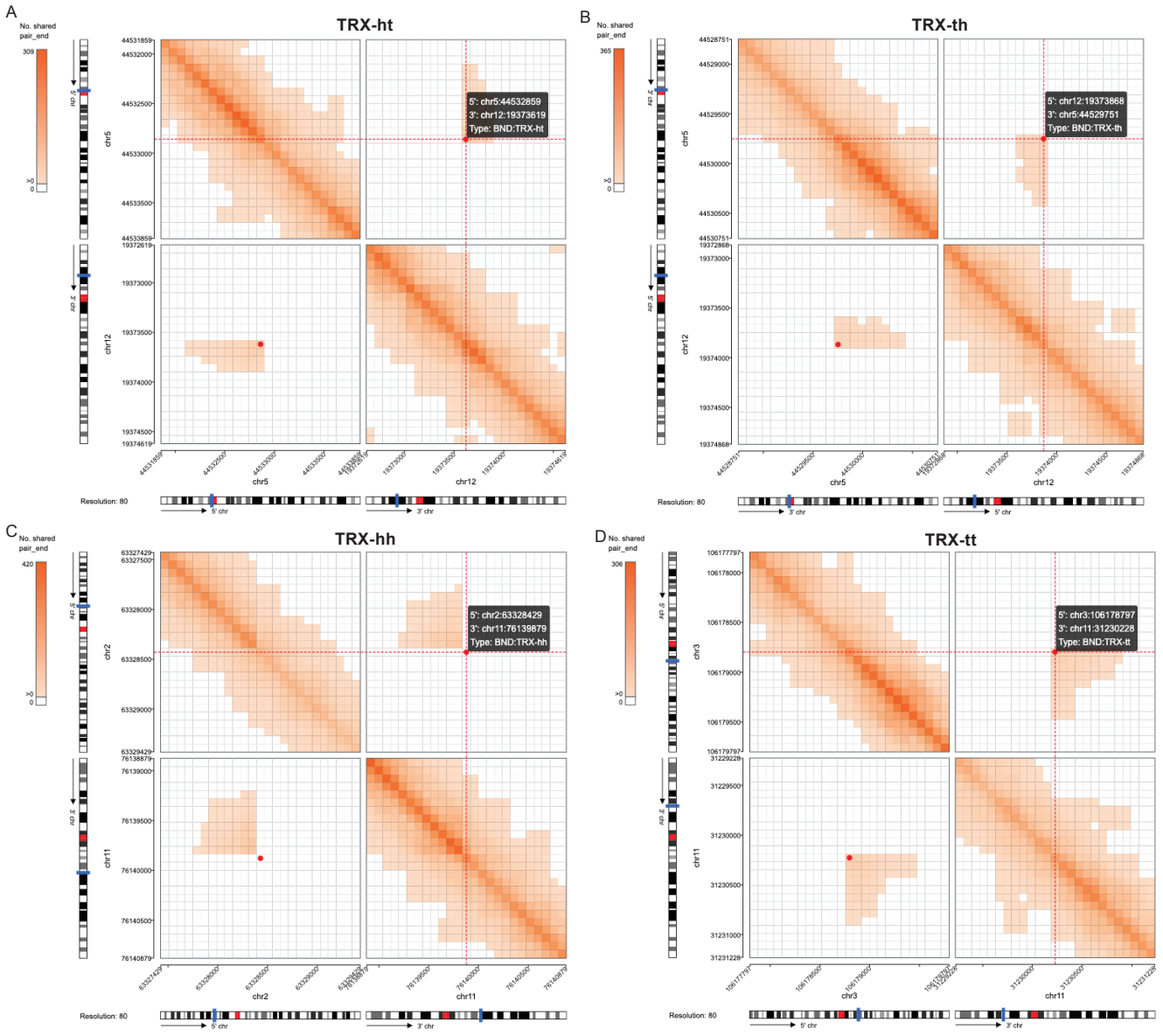
Supplementary Figure S2. Demo representation of the 'CNV Focal Cluster' visualization in single chromosome view. The *Q*-values at each genomic locus are plotted along the vertically arranged chromosome chr6 with green lines representing the cut-off. Cytobands will show tooltips when mouse pointed.

Supplementary Figure S3



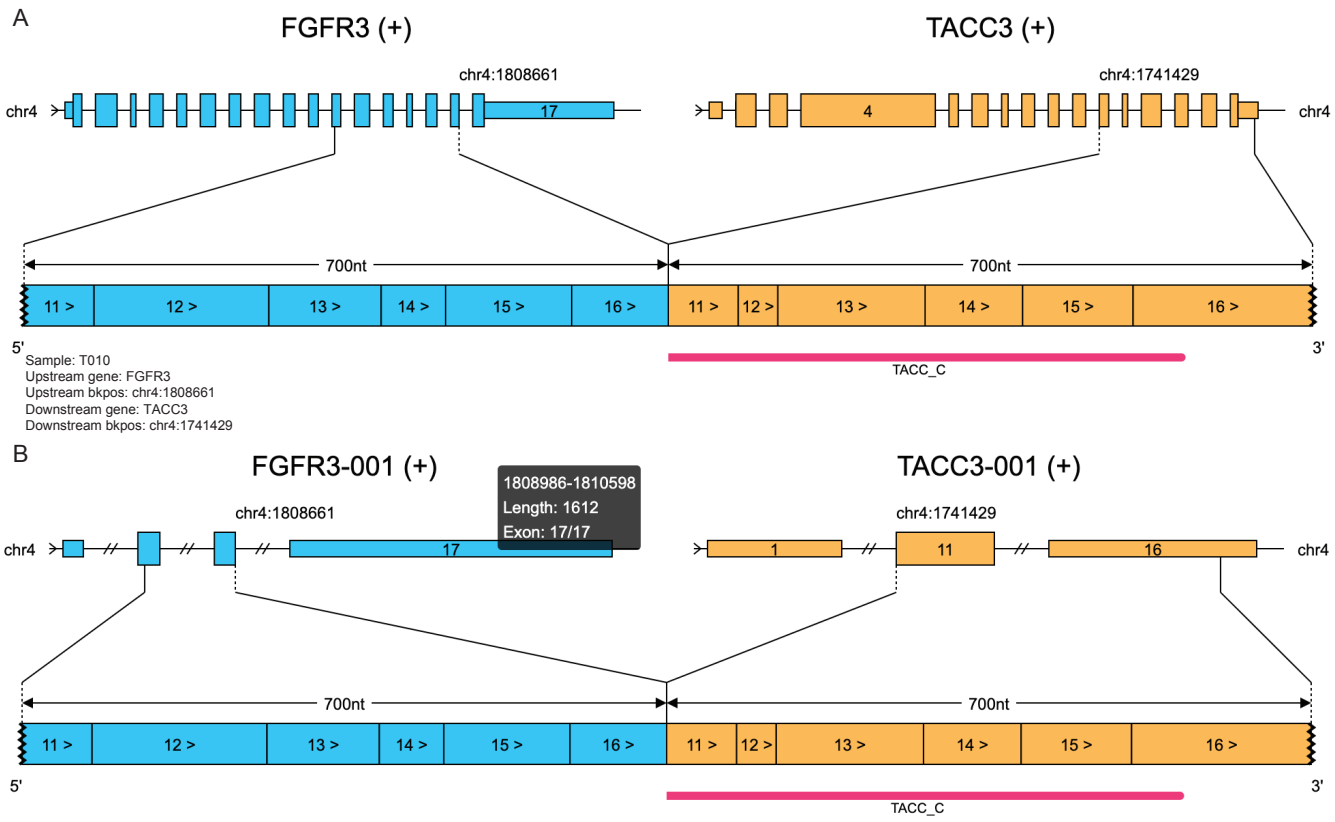
Supplementary Figure S3. Demo representation of the 'SV Heatmap' visualization for different intra-chromosome SV types. (**A**) DEL-ht. (**B**) DUP-th. (**C**) INV-hh. (**D**) INV-tt. DEL, deletion; DUP, duplication; INV, inversion; h, head; t, tail.

Supplementary Figure S4



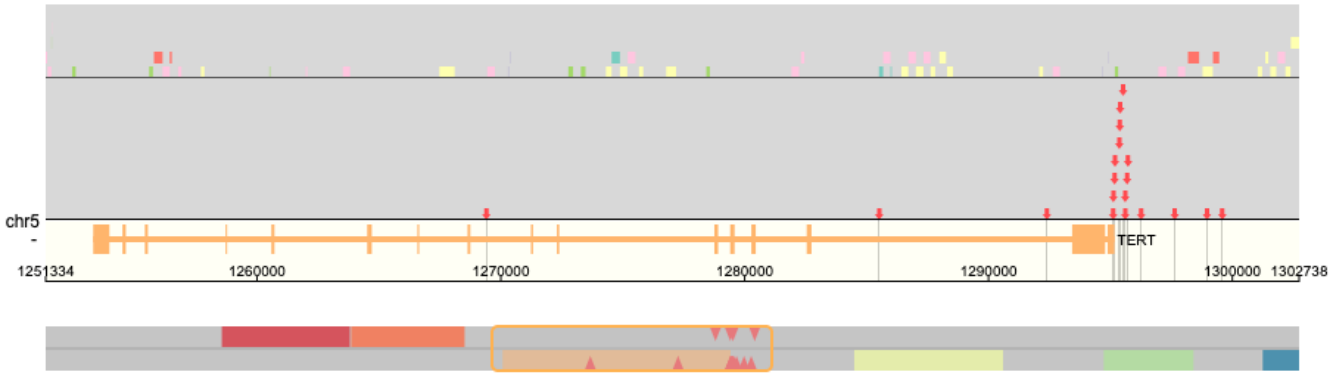
Supplementary Figure S4. Demo representation of the 'SV Heatmap' visualization for different inter-chromosome SV types. (**A**) TRX-ht. (**B**) TRX-th. (**C**) TRX-hh. (**D**) TRX-tt. TRX, translocation; h, head; t, tail.

Supplementary Figure S5



Supplementary Figure S5. Demo representation of the 'Fusion Trans Junction' visualization. Fusion gene *FGFR3-TACC3* is displayed in (**A**) Full transcript mode and (**B**) simplified transcript mode, respectively. Transcript body of gene partner is denoted at the top track by separated exons which have additional tooltips. Fusion junction segment is arranged at the bottom track with relevant exons and bilateral sizes. Protein domains are shown with junction segment. Note that the transcript name is also enabled in the simplified mode.

Supplementary Figure S6



Supplementary Figure S6. Demo representation of the 'Virus Integ HotSpot' visualization with repeatmasker annotations. Genomic elements from repeatmasker are denoted by colored blocks at the top track. All blocks have tooltips to show details.