# Manual of croSSRoad
# (Both CLI and GUI)

## About croSSRoad

**croSSRoad** is a powerful and user-friendly pipeline designed for comprehensive cross-genome comparison among multiple strains of an organism at the Simple Sequence Repeat (SSR) level. It enables users to compare multiple genomes that are linear single chromosomes across all prokaryotes, viruses and eukaryotes, allowing efficient identification of both similarities and differences in SSR patterns across datasets.

With its easy installation, rapid runtime, ease of use, and minimal dependencies, croSSRoad is ideal for large-scale genomic analyses. It provides valuable insights into:

- SSR conservation across diverse genomes,
- Detection of hotspot genes showing significant variation in SSR lengths,
- Chronological tracking of SSR changes to uncover evolutionary trends over time.

Additionally, croSSRoad generates interactive plots for intuitive data visualization and can identify conserved SSR loci along with their flanking regions, which can serve as robust molecular markers or can be used for primer design.

## Features of croSSRoad

- **SSR count in each genome**
  Calculates and reports the total number of SSRs identified in every genome, enabling comparative quantification
- **Relative abundance and relative density**
  Measures SSRs frequency and density relative to genome size to assess SSR richness and distribution

- **Conserved, shared, and unique SSR motifs**
  Identifies SSR motifs that are conserved across genomes, those shared by a subset, and those unique to individual genomes
- **SSRs with conserved flanking regions**
  Detects SSRs flanked by conserved sequences, ideal candidates for designing molecular markers or primers
- **Location of SSR in the genome**
  Maps the precise genomic locations of SSRs, distinguishing between coding, non-coding, and intergenic regions
- **SSR variation in genes**
  Highlights genes containing variable SSRs, which may be functionally relevant or under selective pressure
- **SSR variation with respect to the time point**
  Traces SSR length changes across genomes sampled at different time points, aiding evolutionary and epidemiological analyses

# 1. Command Line Interphase

## 1.1 Installation

Installation of croSSRoad can be done easily through mamba, We recommend users to make a new environment to install croSSRoad

Steps to follow :

- `$conda create -n env_name`
- `$conda activate env_name`
- `$conda install conda-forge::mamba`
- `$mamba install -c jitendralab -c bioconda -c conda-forge crossroad -y` OR

```
$mamba install crossroad -c jitendralab -c bioconda -c
conda-forge --channel-priority flexible -y
```

To update croSSRoad version

```
$mamba update -c jitendralab -c bioconda -c conda-forge
crossroad
```

## 1.2 Usage

To use croSSRoad run

```
$crossroad -h
```

Run the main croSSRoad analysis pipeline.

─ Options

| | | |
|---|---|---|
| --install-completion | | Install completion for the current shell. |
| --show-completion | | Show completion for the current shell, to copy it or customize the installation. |
| --help | -h | Show this message and exit. |

─ Input Files (provide --input-dir OR --fasta)

| | | | |
|---|---|---|---|
| --input-dir | -i | PATH | Directory containing: `all_genome.fa`, `[genome_categories.tsv]`, `[gene.bed]`. Exclusive with `--fasta`. |
| --fasta | -fa | PATH | Input FASTA file (e.g., `all_genome.fa`). Alternative to `--input-dir`. |
| --categories | -c | PATH | Genome categories TSV file. Optional if using `--fasta`. Ignored if `--input-dir` is used (looks for `genome_categories.tsv` inside). |
| --gene-bed | -b | PATH | Gene BED file for SSR-gene analysis. Optional. If `--input-dir` is used, it looks for `gene.bed` inside. |

─ Analysis Parameters

| | | | |
|---|---|---|---|
| --reference-id | -ref | TEXT | Reference genome ID for comparative analysis. Optional parameter for reference-based comparisons. |

--output-dir -o DIRECTORY Base output directory for the job. [default: jobOut]

--flanks -f Process flanking regions.

─ PERF SSR Detection Parameters

--mono INTEGER Mononucleotide repeat threshold. [default: 12]

--di INTEGER Dinucleotide repeat threshold. [default: 6]

--tri INTEGER Trinucleotide repeat threshold. [default: 4]

--tetra INTEGER Tetranucleotide repeat threshold. [default: 3]

--penta INTEGER Pentanucleotide repeat threshold. [default: 3]

--hexa INTEGER Hexanucleotide repeat threshold. [default: 2]

─ Filtering Parameters

--min-len -l INTEGER Minimum genome length for filtering. [default: 1000]

--max-len -L INTEGER Maximum genome length for filtering. [default: 10000000]

--unfair -u INTEGER Maximum number of N's allowed per genome for Crossroad analysis. [default: 0]

--min-repeat-count -rc INTEGER Minimum repeat count for hotspot filtering (keeps records > this value). [default: 1]

--min-genome-count -g INTEGER Minimum genome count for hotspot filtering (keeps records > this value). [default: 2]

─ Performance & Output

--threads -t INTEGER Number of threads for Crossroad analysis. [default: 50]

--plots -p Enable plot generation.

--intrim-dir TEXT Name for the intermediate files directory (within the main job output dir). [default: intrim]

## 1.3 Default Run

### a) Reference-free analysis

```
$croSSRoad -i <input_directory> -p
$crossroad  -fa <multifasta genome.fa>  -c <metadata.tsv> -b
<gene.bed> -p
```

**b) Reference-based analysis**

```
$croSSRoad -i <input_directory> -p -ref <ref ID>
$crossroad  -fa <genome.fa>  -c <metadata.tsv> -b <gene.bed> -p
-ref  <ref ID>
```

## 1.4 Input File Preparation

The pipeline requires three input files, formatted as shown in the provided test data:

**a) Multi-FASTA File**

- Provide a single multi-FASTA file containing all genome sequences.
- If you have a reference genome, include it in this same multi-FASTA file.
- Note : ensure that all genome sequences are in one direction only. For more information see section “Orientation detection”

**b) Metadata File**

This should be a tab-separated file with exactly four columns:

| genomeID | category | optional_category | year |
|---|---|---|---|
| EPI_IS_023 | A.1 | USA | 1994 |
| EPI_IS_024 | B.1 | USA | 2003 |
| EPI_IS_025 | A.1 | India | 2019 |

Note: The column headers are fixed and must not be changed. Only the third column header can be changed according to data for example. Here “optional_category” can be replaced with “country”

- category:

- This column is essential and is used to group genomes for downstream analysis and plotting.
- Use meaningful grouping criteria such as *lineage*, *cluster*, *sample type*, or *source of collection*.
- If your dataset has fewer than 15 genomes and you wish to view each genome individually (without grouping), you can repeat the genomeID values in the category column.

- optional_category:

  - Use this column for descriptive information that is not used for grouping.
  - This category name is dynamic and users can enter any name to this category that will be automatically visible throughout the output data.
  - For example: *country*, *continent*, or *sample source*.

Important: If your dataset contains more than 15 genomes, some category must be used to group them. Otherwise, the resulting plots will be difficult to interpret.

**c) BED File (Gene Annotation)**

Provide a BED-format file without a header, containing four columns:

| genomeID | Start | Stop | Gene name |
|---|---|---|---|
| EPI_IS_023 | 210 | 700 | OPX_2 |
| EPI_IS_023 | 1000 | 1800 | PHOI |
| EPI_IS_023 | 2200 | 3000 | FOXO |

- This file should include the gene start and stop positions for each genome.
  Important Note : BED file should be tab separated and should be sorted in ascending order. Negative integers not allowed. The start value should be bigger than the stop value.

## 1.5 Reference-Based Comparative Analysis

To perform reference-based comparative analysis, use the flag:

```
-ref <reference genome ID>
```

- Replace <reference_genome_ID> with the exact name of the reference genome as it appears in the multi-FASTA file, metadata file (metadata.tsv), gene annotation file (gene.bed)
  Important: Do not provide the reference genome details separately.
  Ensure the reference is included within all three input files, just like the other genomes.

## 1.6 SSR Repeat Types: Mono to Hexa

The pipeline uses PERF for SSR (Simple Sequence Repeat) identification.

- By default (when no specific flags are provided), the pipeline automatically detects SSRs using the following minimum repeat thresholds:
  Mono: 10, Di: 6, Tri: 4, Tetra: 3, Penta: 2, Hexa: 2
- If you wish to customize the minimum repeat count for any motif type, you can specify your own values using the appropriate flags.

Note: Custom parameters allow fine-tuning the sensitivity of SSR detection based on your dataset or research needs.

## 1.7 Output Files

After running the pipeline, a directory named jobOut will be automatically created. This folder serves as a central location to store all job outputs and helps keep your work organized.

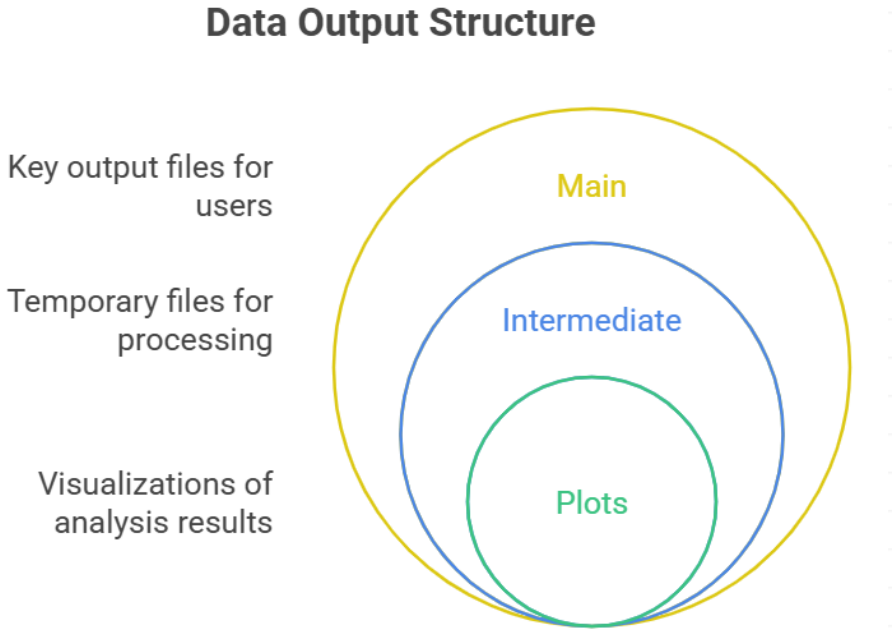
Inside jobOut, a subdirectory named job_XXXX will be generated for each run, where XXXX is a unique job number. Each job_XXXX directory contains:

- A log file with run information

- An output directory structured into three main subfolders:
    - Main
    - Intermediate
    - Plots

If the user wishes to specify the output directory name, can use the flag

```
-o <output_directory_name>
```




**a) Main Directory**

The Main folder contains the key output files that are directly useful to the user:

- mergedOut.tsv: file consists of all the SSR information present in all compared genomes related to SSR motif,length, position, repeat count, GC % of SSR and genome, N count, all metadata information and length of genome.

- ssr_genecombo.tsv: file contains information only of SSR present in genes. This file is a combined file of mergedOut.tsv and gene.bed. Also include position of SSR in gene
- hssr_data.csv: file consist of genes which are showing SSR polymorphism in detail
- mutational_hotspot.csv: file consists of all the hotspot genes with SSR polymorphism in a concise way.

These files summarize important results such as SSR locations, gene associations, and hotspot regions. In the main folder one more folder will appear with the name "Flanks". See section "Flanking Region"

**b) Interim Directory (Intermediate directory)**

The Intermediate folder holds all supporting and intermediate files generated during processing. These are not usually required for downstream analysis but may be useful for troubleshooting or advanced use:

- all_variations.csv : file formed before mutational_hotspot.csv. Consist of all type of mutations i.e. point mutations, SSR length and number variation
- filtered_genomes.fa: contain all genomes passing quality filtered criteria
- intersect_output.bed: file formed before ssr_genecombo.tsv for efficient processing
- mh_tmp.csv: file formed before final mutational_hotspot.csv for proper processing
- perf_out.tsv: contain PERF tool output
- repeatvariation.csv: formed before final mutational_hotspot.csv for proper processing
- clean_genome.fa: illegal characters replace to N
- genome_stats.tsv : contains length of genome and N's in the nucleotide sequence
- locicons.txt : contains all the SSR which are conserved across all genomes
- no_overlap_genes_output.bed : contains genes which are not overlapping with SSR
- perf_params.txt : file consist of parameter used for mono to hexa by the user
- cyclical_variation.csv : formed before final mutational_hotspot.csv for proper processing
- hotspot_single_matches.csv: formed before final mutational_hotspot.csv for proper processing
- mergedOut.bed: file formed after mergedOut.tsv for overlapping with genes
- no_overlap_output.bed : mergedOut.tsv not consisting of SSR in genes

- reformatted.tsv: file formed after perf_out.tsv for further processing
- genes_non_ssr.tsv : file contains gene names without SSR
- ssr_non_gene.tsv: file which contains mergedOut.tsv SSR rows which does not fall in gene

**c) Plots Directory**

The Plots folder contains 14 subdirectories, each corresponding to a different visualization:

- category_country_sankey
- gene_country_sankey
- ssr_conservation
- motif_conservation
- motif_repeat_count_by_gene
- ssr_gene_intersection
- ssr_gc_distribution
- temporal_ssr_distribution
- upset_plot
- relative_abundance
- relative_density
- motif_distribution_heatmap
- ssr_gene_genome_dot_plot
- reference_ssr_distribution (obtain during reference based analysis)

Each subdirectory includes:

- Plots in PNG and interactive HTML format
- The CSV data used to generate the plots
- A statistics file summarizing each plot

The HTML versions of the plots allow for interactive exploration of the data.

## 1.8 Flanking Region

Use the `-f` flag to extract 10 bp flanking regions on both sides of each SSR, which can serve as potential primer sites. The complete output will be stored in the main directory under a folder named flanks. Within this folder, pattern_summary.tsv lists conserved SSRs along with their flanking sequences. Users can perform an online BLAST search to check for species-specificity of these regions. For primer design, the full coordinates—including 10 bp upstream flank, SSR, and 10 bp downstream flank—are available in flanked.tsv.

## 1.9 Orientation Detection

The croSSRoad pipeline uses PERF tool to detect SSRs in a single direction, as it analyzes genome sequences provided in one strand only (typically from a multifasta file). Consequently, SSRs are identified in just one orientation. If all genome sequences are oriented the same way, motif comparisons remain consistent. However, if the sequences are in mixed orientations, different SSR motifs may be detected across genomes, leading to misleading or invalid comparisons.

To detect orientation of genome there are few possible ways:

1. Use blast considering “strand” in outfmt format
2. Use minimap
3. Use seqkit sliding
4. Use mummer

# 2. Graphical User Interphase (GUI)


## 2.1 Accessing the Graphical User Interface (GUI) of croSSRoad

To access the GUI of croSSRoad, open the following link:
https://crossroad.igib.res.in/

## 2.2 Running croSSRoad

Click “Run Analysis Now” or “Start Analysis” at the top right corner. You will then see an option to upload input files.

### 2.2.1 Preparing Input Files:

Required input files:

- Multifasta file (.fa; minimum two genomes) – mandatory
- Metadata file (.tsv) – optional
- BED file (.bed) – optional

To view File format:

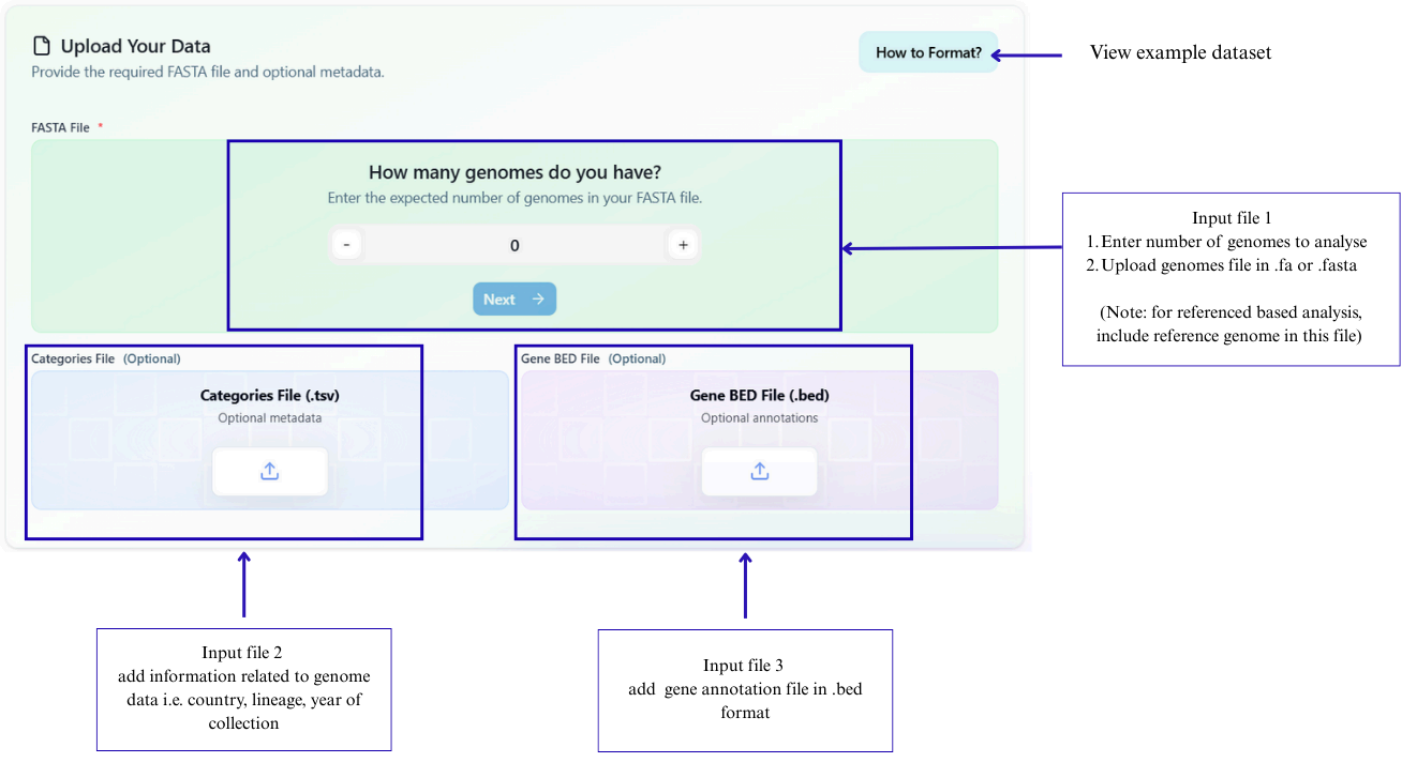
- Refer to the examples provided in the navigation movable navigation bar section, or click on “How to Format” (available at the top of the upload option) to view sample datasets.
- Detailed guidelines can also be found in Section 1.4: “Input File Preparation” of the documentation.
- Example dataset here consist of 404 genomes in multi-fasta, metadata information and gene positions for all the genomes.

**Note:**
If reference-based genome comparison is required, include the reference genome sequence in the multifasta file itself.

### 2.2.2 To Upload Files

Steps to upload the file:

1. **Upload FASTA file (mandatory):**
    - Provide a .fa or .fasta file containing at least two genomes.
    - Enter the total number of genomes included in the file.
    - Note: For reference-based comparative analysis, include the reference genome in this FASTA file.
2. **Upload metadata file (optional):**
    - Upload a .tsv file containing additional information such as country, lineage, and year of collection.
3. **Upload gene annotation file (optional):**
    - Upload a .bed file containing gene annotation details.
4. **Check formatting examples:**
    - Click “How to Format?” or view the provided example datasets to ensure correct file preparation.




### 2.2.3 Set the Configuration

This section allows users to customize the detection parameters for SSR motifs. The default values are pre-optimized, but advanced users can modify them as needed. The configurable parameters are:

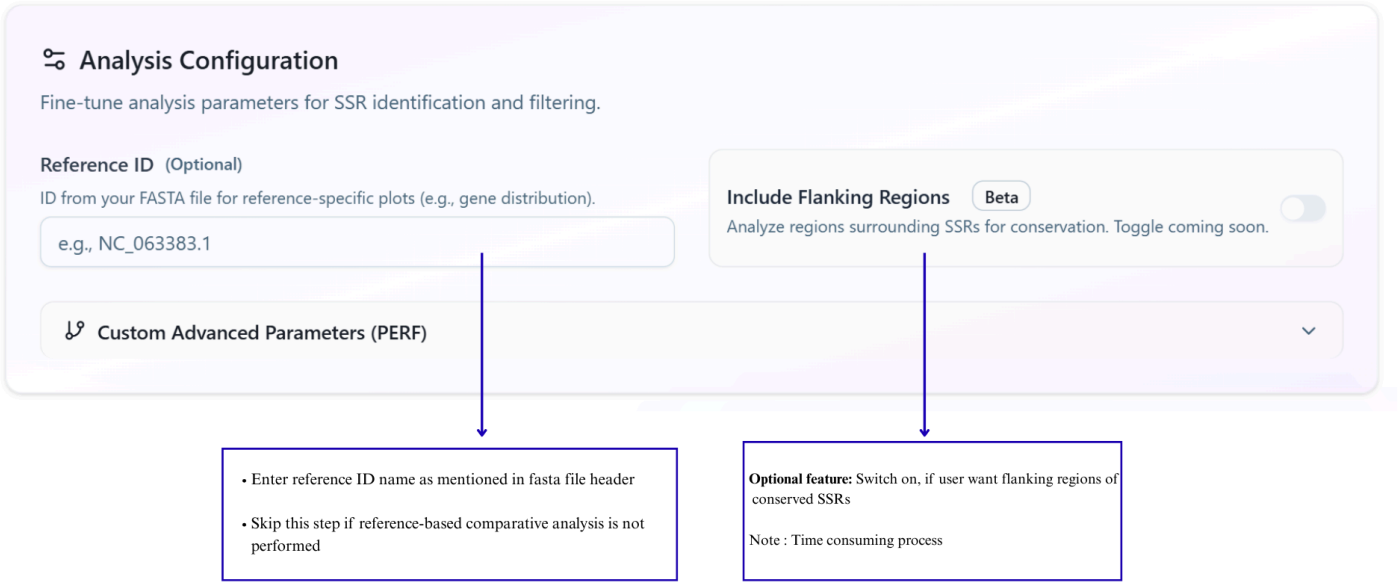
1. **Repeat thresholds for SSR motifs:**
    - Mono: Minimum number of mononucleotide repeats (default: 12).
    - Di: Minimum number of dinucleotide repeats (default: 6).
    - Tri: Minimum number of trinucleotide repeats (default: 4).
    - Tetra: Minimum number of tetranucleotide repeats (default: 3).
    - Penta: Minimum number of pentanucleotide repeats (default: 3).
    - Hexa: Minimum number of hexanucleotide repeats (default: 2).
2. **Sequence length filters:**
    - MinLen: Minimum sequence length considered for analysis (default: 1000 bp).

- MaxLen: Maximum sequence length considered for analysis (default: 10,000,000 bp).

3. **Other parameters:**

   - Max N Bases: Maximum allowable ambiguous bases ("N") in the sequence (default: 0).
   - Thread: Number of parallel threads to be used for computation (default: 50).
   - Min Repeat Count: Minimum total number of repeats required to report an SSR polymorphism (default: >1).
   - Min Genome Count: Minimum number of genomes supporting SSR polymorphism in gene (default: >2).

**Note:** Users unfamiliar with SSR filtering are advised to use the default values. Only change minimum length and maximum length to prevent exclusion of any genome.

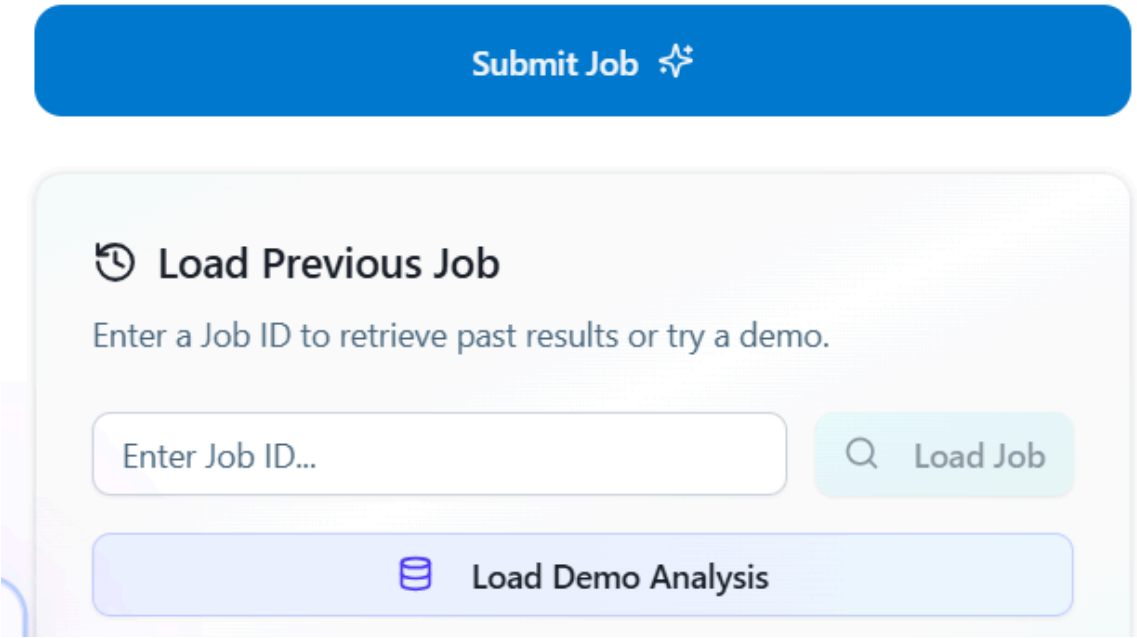



### 2.2.4 Submit the Job

- Once the required files have been uploaded and the parameters adjusted as needed, the user can submit the job by clicking on the Submit Job button (see Figure below ).

- After successfully submitting a job, make sure to save the Job ID. This ID will be required to retrieve the results at a later time.
- If you wish to view results using the example dataset, click on the "Load Demo Analysis" option.

Both the Submit Job and Load Job/Demo Analysis options are located on the right-hand side of the page.



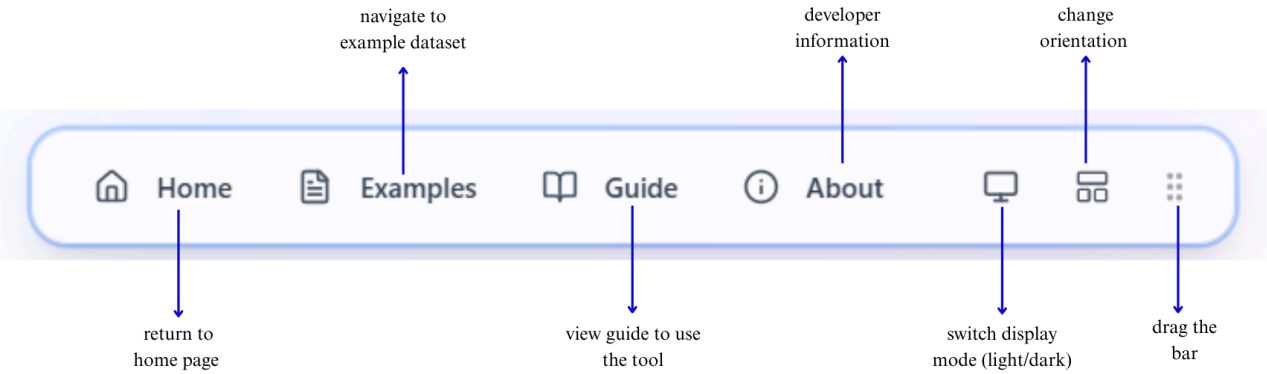

## 2.3 Other Features

1. **Movable Navigation Bar**

   The navigation bar (see Figure X) provides quick access to different features of the croSSRoad tool:

- Home – Return to the home page.
- Examples – Navigate to example datasets for practice or demonstration.
- Guide – Open the user guide for instructions on using the tool.
- About – View developer and project-related information.
- Switch Display Mode (Light/Dark) – Toggle the display theme between light and dark mode.
- Change Orientation – Adjust the layout orientation of the interface.

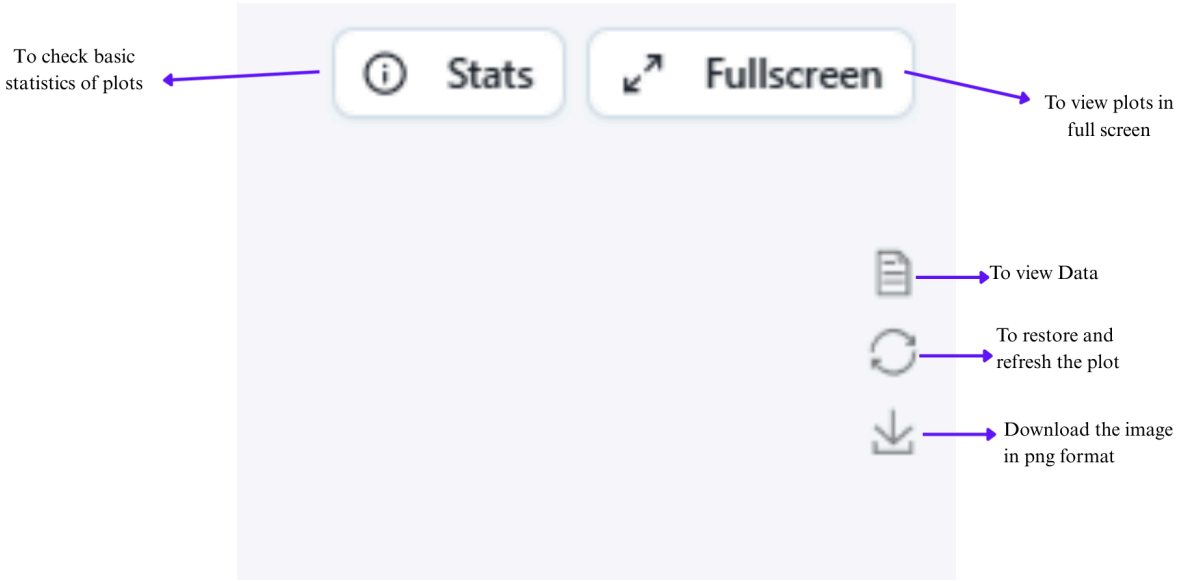
- Drag Bar – Allows you to move the navigation bar as per your convenience.




## 2. Features in plots

Each plot generated in the analysis interface includes a right-hand panel with interactive options to help users explore, restore, and save their visualizations.

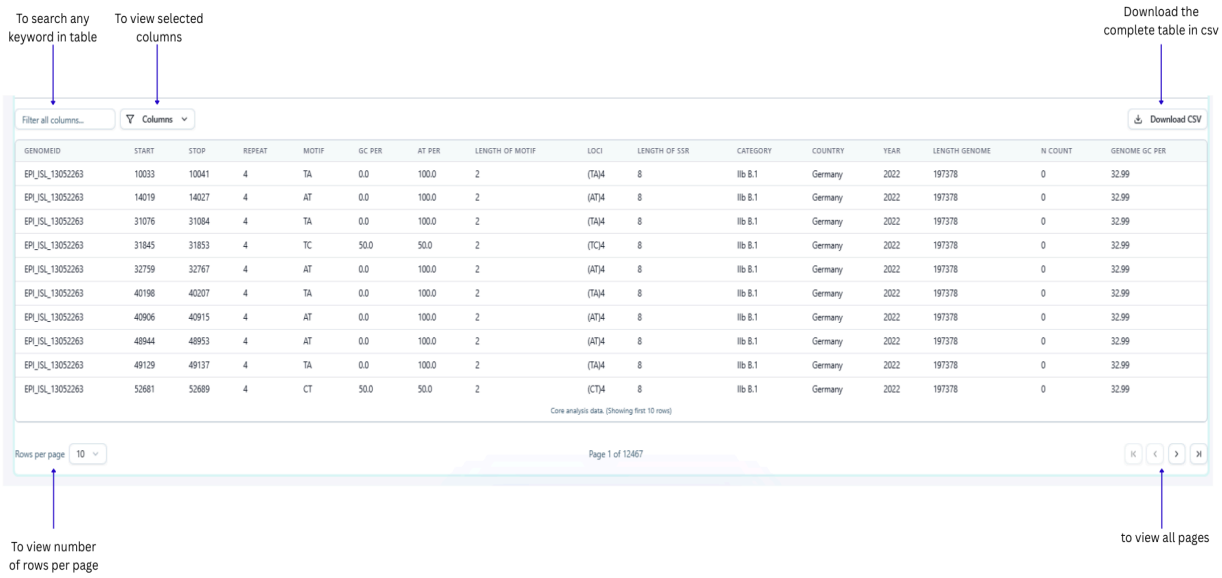
- Stats: Displays the basic statistics of the selected plot.
- Fullscreen: Expands the plot to fullscreen mode for better visualization.
- View Data: Opens the underlying dataset used to generate the plot.
- Refresh: Restores and refreshes the plot to its default view.
- Download: Exports the plot as a PNG image file.
- Plots can be zoomed in and out using mouse cursor




### 3. Interactive feature for data tables

Each plot in the results section is accompanied by a detailed data table. The table provides flexibility for filtering, customizing, and exporting the underlying dataset.

- Search bar: Allows keyword-based filtering within the table.
- Columns: Enables selection of specific columns to display.
- Rows per page: Controls the number of rows visible per page.
- Pagination controls: Navigate across pages to view the complete dataset.
- Download CSV: Exports the complete table in CSV format for offline use.




## 2.4 Interpreting Example Dataset

To explore the example dataset, click on “Load Demo Analysis”. This will generate the results that demonstrate how croSSRoad presents different types of outputs.

The Result Section is organized into four main parts. Each part includes plots for visualization and corresponding data tables displayed below the plots.

### 1. Core Data and Plots

This section contains data equivalent to the CLI output file mergedOut.tsv

- **Category → Country Sankey Plot**
  Visualizes the distribution of genomes across categories and countries.

- **SSR GC Distribution Plot**
  Displays SSR counts for each genome, with GC content represented by color (as indicated in the legend).

- **Motif Conservation Plot**
  Pie chart showing conserved and unique motifs across all genomes.

- **SSR Conservation Plot**
  Similar to motif conservation but based on SSR length instead of motif identity.

- **UpSet Plot**
  Displays the number of motifs conserved across categories.

- **Relative Abundance Plot**
  Stacked bar chart showing SSR counts across classes (monomer, dimer, trimer, tetramer, pentamer, hexamer) in each category.

- **Relative Density Plot**
  Stacked bar chart showing SSR lengths across classes (monomer, dimer, trimer, tetramer, pentamer, hexamer) in each category.

- **Motif Heatmap**
  Heatmap representing genomes (rows) versus motif counts (columns).

**2. SSR–Gene Intersection**

This section contains data equivalent to ssr_genecombo.tsv

- **Intersection Plot**
  Displays cumulative SSR positions in all genomes against all genes, highlighting whether SSRs occur at gene start, stop, or within the gene body.

- **Reference SSR Count Table**
  Provides SSR counts at positions (IN, intersect_start, intersect_stop) in the reference genome only.

**3. Hotspot Data and Plot**

This section contains data equivalent to mutational_hotspot.csv, depicting SSR polymorphism or repeat count variation within genes across genomes.

**4. HSSR Data and Plots**

This section contains data equivalent to hssr_data.csv

- **Gene–Country Association Plot**
  Sankey plot showing categories and genomes with polymorphic SSRs in specific genes.

- **Temporal Variation Plot**
  Displays SSR length variation over time for hotspot genes.

- **SSR Dot Plot**
  Plots polymorphic motifs as dots for each gene and genome; dot size represents SSR length.