Chapter 17
Interactive Data Analyses Using TBtools

Chengjie Chen and Rui Xia

Abstract Increasing biological data provide us unprecedented ability to uncover the mystery of life. To leverage oncoming large biological data, efficient and effective data analysis is indispensable for biological research. However, data analysis has become a major challenge to biologists, most of who are not skillful in computer science, thereby limiting the utilization efficiency of biological data. Although a lot of bioinformatics software have been developed in the community, the majority of them require users to work under command-line environments or even be familiar with programming languages, with few of them focusing on freeing users from elaborate command-line-based tasks.

Here, we present TBtools, an out-of-box solution to routine biological data analyses. The toolkit integrates ~150 practical functions for data analyses and visualization, with a user-friendly graphical interface. In this chapter, we describe the design philosophy, development objectives, and main characteristics of TBtools. We also provide a comprehensive introduction of its main functions, especially those included in the "Sequence Toolkits" and "Graphics" catalogs, and advanced features, like R plugins that contributed by senior users. A few practical tutorials are presented to demonstrate the superb functionalities and outstanding interactive nature of TBtools.

Keywords TBtools · Bioinformatics · Function integration · Data analysis · Data visualization

17.1 Design Philosophy and Development Objectives

In recent years, high-throughput sequencing technologies have been developing rapidly in the field of life sciences, and big data analyses have become an indispensable part of biological research. For the vast majority of biologists, the effective

C. Chen · R. Xia (✉)
College of Horticulture, South China Agricultural University, Guangzhou, Guangdong, China
e-mail: rxia@scau.edu.cn

M. Chen, R. Hofestädt (eds.), *Integrative Bioinformatics*,
https://doi.org/10.1007/978-981-16-6795-4_17

use of these data is not only an opportunity, but also a challenge. For instance, through genome-wide association analysis with population resequencing and large-scale trait surveys, we can identify key genes associated with certain important traits efficiently. Constructing gene co-expression networks from transcriptional expression profiles, we can screen out potential hub regulatory factors of key traits and steer the direction of further gene function studies. Most of these analyses often involve two stages, which we simplified as upstream and downstream data analyses, and for both of them, researchers are required to have two major aspects of knowledge, computer science and biology. Upstream data analyses often require large-scale data operations, which run on high-performance servers and consume a lot of computation resources, such as profiling whole-genome SNP sites from resequencing data with data size of >10 TB, or calculating gene expression levels from >100 GB transcriptome sequencing data. These analyses are common procedure for different projects, mainly involving large data calculations and little relevance to specific biological questions. Many powerful bioinformatics software or tools have been developed to meet this type of common demand. Often commercial service providers can be relied on this part of analyses, alternatively, researchers can learn to use these tools in a project and reuse them repeatedly for different projects. In contrast, downstream data analyses are much more complicated and need "personalized recipe" of analyses to solve various biological questions. Normally there is no routine way to follow for these analyses, including various small tasks of different purposes, such as conversion of all kinds of file formats, sorting and extraction of certain text information, representation of distinct results, etc. To handle these small and specific analyses, researchers are often required to search, test, and learn to use a large number of tools (commands or tools composed in different programming languages) for different functions or to program them into different pipelines to achieve certain analysis goals. And in most cases, this process of searching, testing, learning is not repeatable among different projects, therefore greatly increases the cost of simple data analyses and reduces the efficiency of scientific research. Based on these observations, since 2015, we have been developing a bioinformatics combo toolkit, TBtools, to integrate hundreds of data analysis functions routinely needed from biological laboratories, for streamlined and simplified usage. This chapter gives an overview and introduction on the development and usage of TBtools.

17.1.1 Development Logic

"Know the tricks of the trade," this idiom fits to the process of learning and mastering any bioinformatics software.

17.1.1.1 Practical First, Concise Utmost

The development of TBtools starts from the demand of daily data analyses of the authors. The realization of bioinformatics functions is designed to fit the needs of daily data analysis in the most practical and simplest way, for example, the routine bulk sequence extraction from fasta sequence files and the simple local BLAST sequence search (Altschul et al. 1997). The design of the software interface follows the rule of "the simpler, the better," only retaining concise prompt message and necessary parameter settings. All the rest, including intermediate files generated automatically are hidden without showing. And many options of certain functions are simplified or automated, such as the automatic recognition of the format of input data (e.g., the automatic selection of BLAST sub-programs based on the query and subject sequence types), automatic file format conversion, and programmed adjustments of file names (such as file names containing spaces or special characters) (Fig. 17.1).

17.1.1.2 The Simple "IOS" Logic

The implementation of each function in TBtools strictly follows the most basic programming logic:

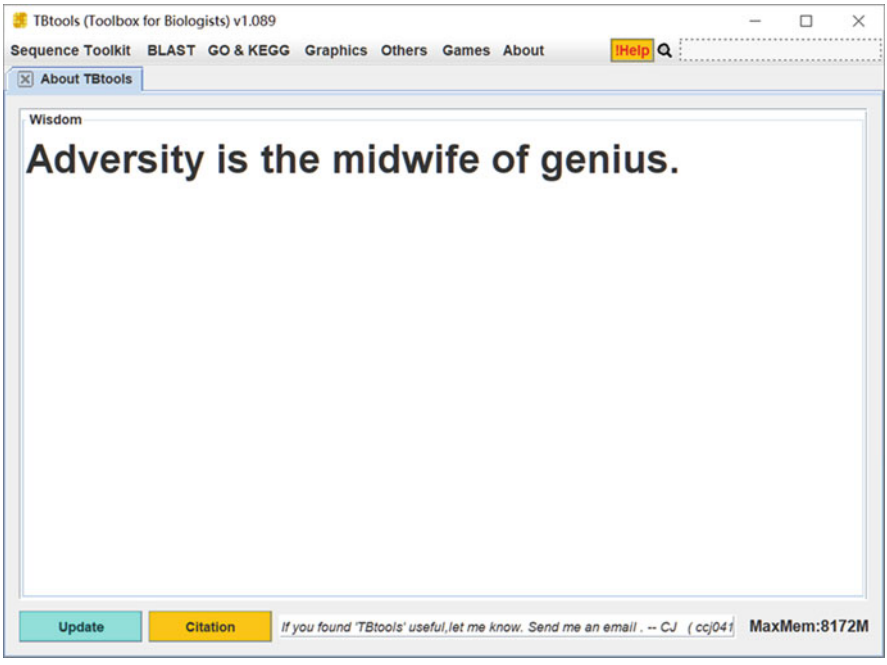

Fig. 17.1 TBtools Main Interface

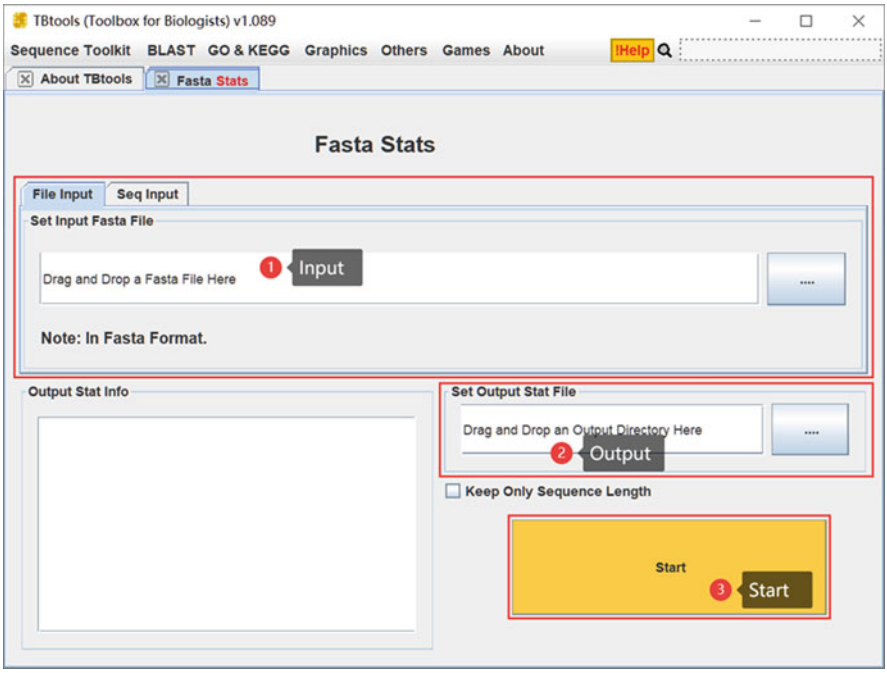

Fig. 17.2 The "IOS" Logic of TBtools Interface

(a) set Input data
(b) set Output data
(c) press "Start" button to proceed

that is, "Input, Output, and Start" logic (Fig. 17.2).

When using any function of TBtools, users only need to set the input and output files, and then click the "Start" button to start the analysis. Worthy of note is that the "Start" button is always bright yellow, which is easy for a quick notice.

17.1.1.3 User-Driven Development

The main driving force for the development of TBtools is real data analysis demands from hundreds of thousands of scientific researchers. Since the first release of TBtools in 2016, there are >50,000 stable users. Users' feedbacks such as suggestions, comments, and new demands are the motives of our continuous development and innovation of the software. The software has a public program repository at Github, https://github.com/CJ-Chen/TBtools/releases, where installers of TBtools can be downloaded and users can report usage issues. We have also established real-time user communication communities in Tencent and Telegram (Fig. 17.3; http://118.24.17.128/TBtoolsUserGroup.png and https://t.me/

Fig. 17.3 TBtools Real-Time User Community

joinchat/Q6WHOhWOlHHOkMgRGoN7nw), and a TBtools user forum (http://www.tbtools.icu:1234). These communities ensure that the development team to communicate with software users in real time, collecting user feedbacks in time for troubleshooting and further development of new features.

17.1.2 Development Objectives

17.1.2.1 One-Click Environment Configuration

In projects dominated by biological questions, bioinformatics is more of a powerful technology. In actual data analyses, users are required to install different software for different needs. For example, when plotting a genome circle map, we need to configure the Perl language environment before installing the Circos software package (Connors et al. 2009). During this period, a lot of source code compilation work is involved, and different software installation problems are often encountered, such as incompatible system platform versions. In fact, installing and configuring software on servers is already a critical step in bioinformatics data analysis that consumes lots of time and energy. Therefore, we have implemented almost all functions of TBtools using pure Java code, ensuring cross-platform characteristics. For a small number of mature software, we have gathered their cross-platform binaries and packaged them into single software installers.

17.1.2.2 Graphics User Interface for Data Analysis

At present, common bioinformatics data analysis tools often require users to be familiar with programming languages such as R language, or familiar with command-line working environments such as DOS or Shell. The learning curve is steep. For most researchers most of whose work is not done on computers, learning costs are too high. It is difficult to master and easy to forget. The initial goal of TBtools development is to enable all users to quickly master the use of the software and start data analyses right out of the box. Although TBtools can also be run through the command line, we focus on creating specific user-friendly interfaces for each useful function.

17.1.2.3 Function Integration

In daily data analyses, users need to use different software in combination to complete a simple analysis task. For instance, to "assess the similarity of two protein sequences in a certain species," we may need to use a text editor or write a script to extract two protein sequences, then use BLAST (local or web) to compare the two sequences, and finally use other software or tools to visualize the results of the comparison. Frequent switching of software takes much time of scientific researchers, and besides, it is easy to interrupt users' thoughts. One of the development goals of TBtools is to fully integrate simple analysis functions. Users can quickly complete sequence extraction, BLAST and direct visualization in TBtools. Covering more than 150 functions, users can integrate them according to their needs, fully meeting other analysis scenarios.

17.1.2.4 Analysis Automation

Although downstream data analyses do not have a similar obvious pattern as upstream data analysis, there are still many simple and repetitive tasks in some analysis tasks. At present, the genomes of more and more species are sequenced and published. Comparative genomics has become a research hotspot. Among them, one common analysis is mining gene collinear blocks. Correspondingly, the most widely used software is MCscan (Wang et al. 2012). Though the need for analysis is common, the use of MCscan requires users to prepare rather cumbersome input files. In short, users should obtain protein sequences of two species, invoke software such as BLASTP for sequence comparison, integrate gene location information, and finally run the MCscan software. During this period, identifier mismatch may be involved, identifier naming system conflicts and file name prefixes are not unified, all of which will cause the final operation to fail or lead to incorrect results. With TBtools, users only need to place genome sequence and gene structure annotation files of two species and click the Start button.

17.1.2.5 Simplify Complex Analysis

Some data analysis tasks are not only cumbersome but also difficult to achieve. The eFP Browser development team proposed for the first time that expression value coloring on a cartoon vividly displays specific gene expression changes (Winter et al. 2007). This has been applied to a few model organisms, which could be found on the corresponding species genome website. At present, omics data is becoming more and more abundant. Cartoon-style heatmaps can be used for research and result display on all other species, and data analysis and result interpretation can be carried out more intuitively. However, eFP Browser is a browser framework, involving knowledge of computer and network configuration, which is almost impossible for scientific researchers with a pure biological background to implement in a short time. Based on similar ideas, TBtools implements a java-based non-dependency eFP graph function. Users only need to prepare a cartoon template, an expression matrix, and a color mapping relationship table to make eFP graphs. Furthermore, compared to eFP Browser, TBtools supports the output of vector graphics to ensure the clarity and interactivity of the final artwork. A similar advance can also be found on the plotting of Circos. On the whole, TBtools enables biological researchers to or even easy to complete a large number of analyses that seemed difficult before.

17.2 Overview of TBtools Functions

17.2.1 Software Acquisition, Update, and Main Interface

17.2.1.1 Software Acquisition

There are two main ways to get TBtools installation.

(a) Download it directly from the software repository at Github, https://github.com/CJ-Chen/TBtools/releases (Fig. 17.4).
(b) Obtain the latest version of the TBtools installation from the user communities at Tencent or Telegram.

Fig. 17.4 Overview of the TBtools Warehouse

17.2.1.2 Software Update

TBtools has been incorporated with background update, real-time update, and automatic update. While running, TBtools will automatically detect the current version in the background and ask the user whether to update to the latest version, ensuring that users can always use the latest version of the software with more comprehensive and more stable functions.

In an unstable network environment, users can directly obtain the TBtools main program (a jar file) elsewhere and then manually update the software in the two following ways.

(a) Go directly to the main directory of the TBtools installation and complete the program update by replacing the main program file (TBtools_JRE1.6.jar) with a newer one.
(b) Enter the "About" catalog from the main menu of TBtools, click "Update via Jar," and select the downloaded .jar file in the pop-up dialog to complete the update (Fig. 17.5).

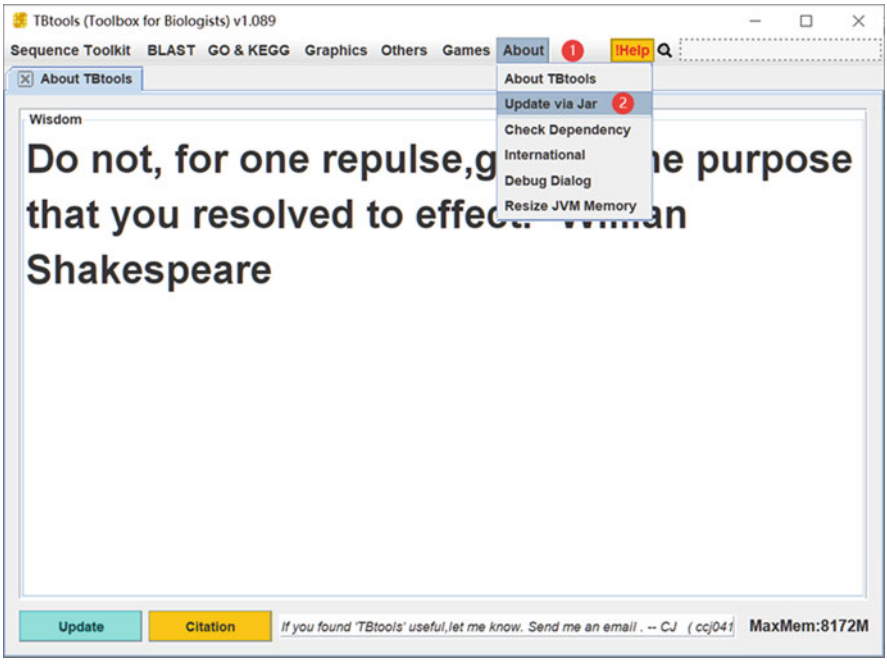

Fig. 17.5 Update TBtools from Main Menu

17.2.1.3 Main Interface of TBtools

After the TBtools software is installed, you can directly run it by a double-click on the program icon from the Start menu. The software will launch and you can see the main interface, which includes several main function catalogs (Fig. 17.6).

1. **Version of TBtools.** It can be used to check whether the current version is the latest one. Version number should be provided when communicate with us for troubleshooting.
2. **Main menu.** TBtools currently divides the main functions into six catalogs (described in detail below).
3. **About Panel.** When TBtools starts, the "About Panel" function will be automatically triggered, and a famous quote will be randomly displayed.
4. **"!Help" button.** When users are confused about a certain function, or do not know how to use or which function to used, they can click this button to get usage examples and related tutorials.
5. **Search box for functions.** There are >150 functions in TBtools. Sometimes it is hard to quickly locate a function level by level through the main menu. A more convenient way is to directly enter keywords for a specific function in the search box (e.g., "Fasta"), and then TBtools will automatically show functions related to

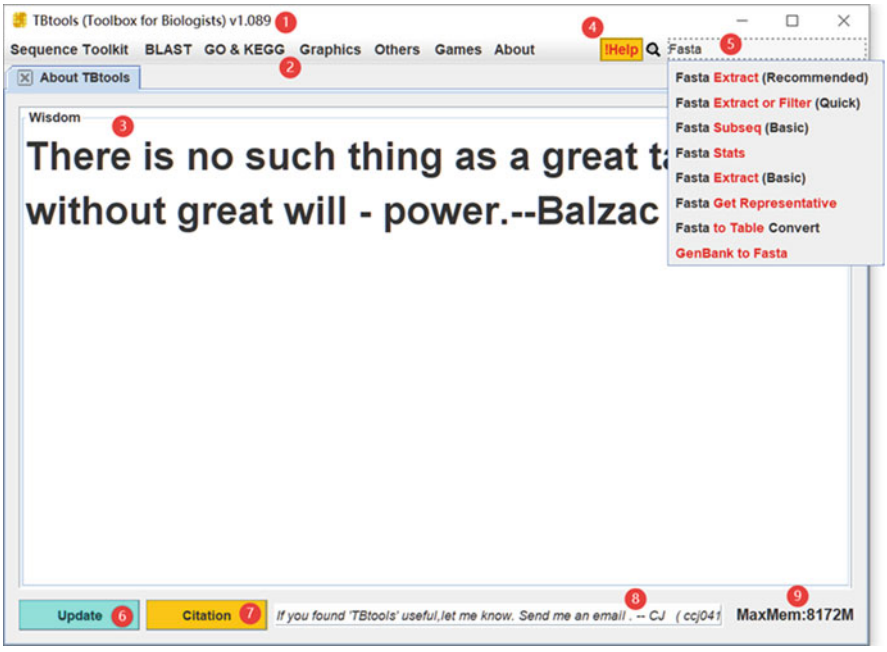

Fig. 17.6 Overview of the Main Interface of TBtools

the keywords (names containing "Fasta") and users can use the function directly by a simply click.
6. **"Update" button.** Users can manually click the "Update" button to force the update of TBtools via background update.
7. **"Citation" button.** This button directs the users to the webpage of the official TBtools publication, for a convenient citation of the software when preparing manuscripts.
8. **Message box.** The message box is used for the users to send their feedbacks (suggestions and comments, or even complimentaries) to developers via email.

17.2.2 Introduction to TBtools Function

17.2.2.1 Overview of Main Functions

The main functions of TBtools are divided into five main catalogs (Fig. 17.7).

1. **Sequence Toolkits**. This catalog mainly includes batch sequence download, bulk sequence extraction, sequence information sorting, and other sequence handling functions. Among them, the "GFF3 Sequence Extract" function is a powerful tool that could be used to extract specific feature sequences based on gene structure annotation information. Users often employ it to obtain the complete set of CDS or gene promoter sequences.
2. **BLAST.** It collects a series of functions from a stand-alone BLAST wrapper, as well as functions for format conversion, result management, and visualization.
3. **GO & KEGG.** This catalog hosts functions for gene set analyses, for example, gene ontology and KEGG pathway enrichment, and for result management and visualization as well.
4. **Graphics**. It covers functions most often used for data representation and visualization, such as venn diagram, heatmap, and seqlogo, as well as a few relatively complex graphs, for example, upset plots and circos.
5. **Others.** Functions that could not be clearly grouped in the previous four catalogs are placed under this one, including functions for text manipulation and phylogenetic analysis.

   In addition to the five main catalogs, TBtools also hosts a few games for pleasure in an extra "Games" catalog, as users may be free when use TBtools for large data analysis which take a long time to process. There is also a "About" catalog which includes options to manually update the software, adjust the maximum available memory for software operation, check whether the dependent programs are complete, view the operation information, etc. (Fig. 17.8).

TBtools currently covers more than 150 functions, and each main catalog has a series of corresponding functions. Limited by the space, we cannot introduced all

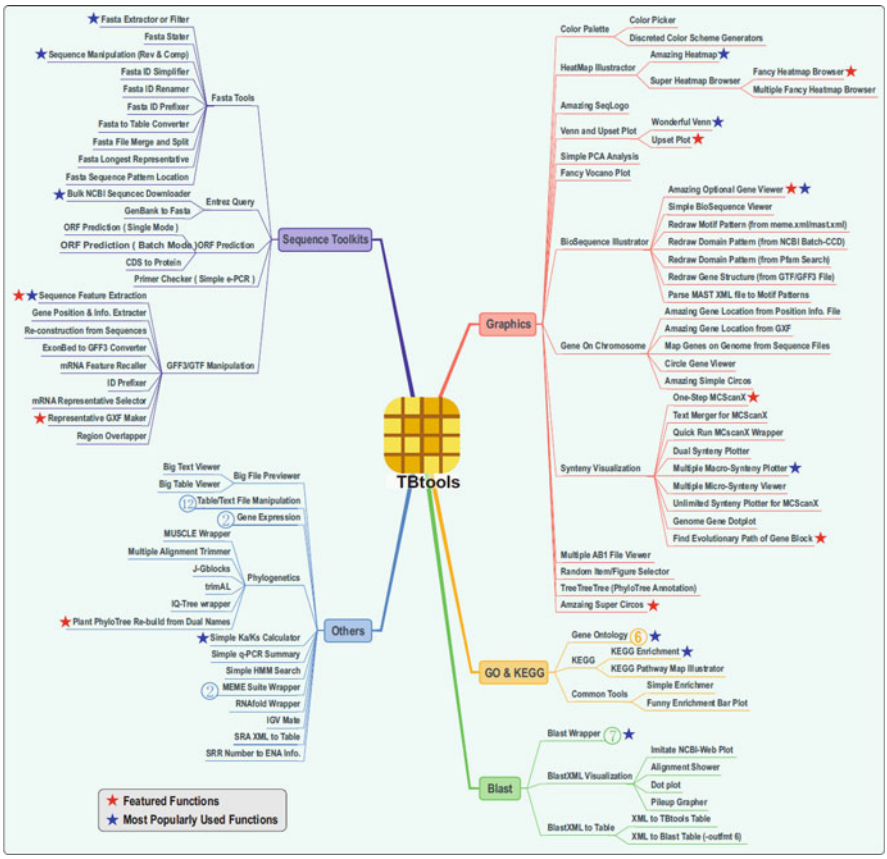

Fig. 17.7 Overview of TBtools Functions (Chen et al. 2020)

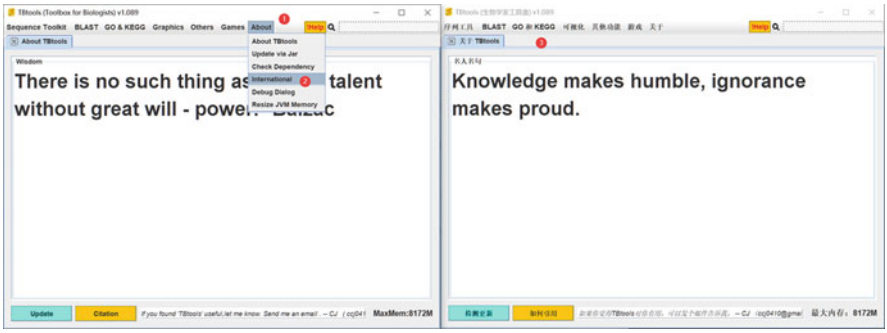

Fig. 17.8 TBtools Interface with Different Languages

of them. The following are the introduction of functions typically used for sequence manipulation and data visualization.

17.2.2.2 Functions for Sequence Manipulation

TBtools contains a large number of functions used for sequence management, mainly for files in Fasta or GFF3/GTF formats (Fig. 17.9).

Sequence Toolkit Functions are divided into five sub-menus.

1. **Fasta Tools**
   (a) **Fasta Extract / Filter.** Batch extract or filter sequence records.
   (b) **Fasta Stat.** Summarize sequence information for a sequence file, such as total sequence length, GC content, etc.
   (c) **Sequence Manipulate.** Manipulate sequences, such as reverse, complement and information sorting.
   (d) **ID Simplify/Rename/Prefix.** Manipulate sequence identifiers, such as, simplify or rename the identifier, or add prefix.
   (e) **Fasta to Table Convert.** Convert sequence file between Fasta and tab-delimited formats.

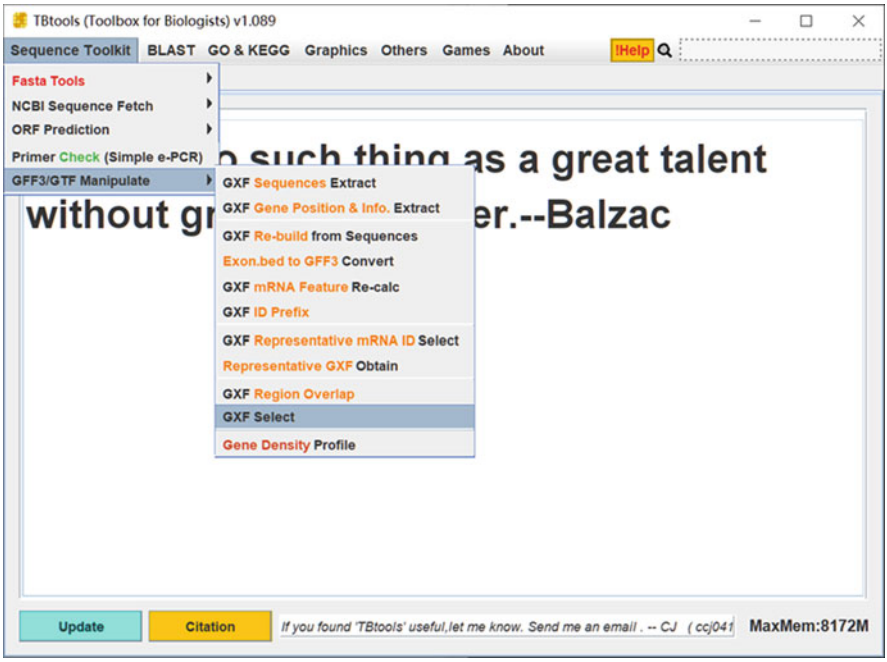

Fig. 17.9 Overview of TBtools Sequence Toolkits

(f) **Fasta Merge/Split.** Merge and split Fasta sequence files.
(g) **Fasta Get Representative.** Extract representative Fasta sequence records.
(h) **Sequence Pattern Locate.** Locate sequence regions that possess specific patterns.

2. **NCBI Sequence Fetch**

(a) **Bulk NCBI Sequence Download.** Download sequences from NCBI using accession numbers.
(b) **GenBank to Fasta.** Convert files from Genbank format to Fasta format.

3. **ORF Prediction**

(a) **Complete ORF Prediction (Single Mode).** Predict complete ORFs of a sequence in six frames.
(b) **Complete ORF Prediction (Batch Mode).** Batch predict the longest ORF in a set of sequences.
(c) **Batch Translate CDS to Protein.**

4. **Primer Check (Simple e-PCR).** Detect all possible amplified fragments for given primers in a specific sequence library.
5. **GFF3/GTF Manipulate**

GFF3/GTF files are standard annotation files storing gene structure information for any give genome sequence files.

(a) **GXF Sequence Extract.** Extract sequences of specific feature from the genome sequence file based on the GFF3/GTF file.
(b) **GXF Gene Posi. and Info. Extract.** Extract gene location and annotation information of a species.
(c) **GXF Re-build from Sequences.** Reconstruct a GFF3 file from given transcript sequences and corresponding reference genome sequences.
(d) **GXF mRNA Feature Re-calc.** Recalculate and add mRNA information to GXF files.
(e) **GXF ID Prefix.** Add specified prefixes to chromosome ID and gene ID.
(f) **GXF Representative mRNA ID Select.** Obtain all representative transcript IDs from GFF3/GTF files.
(g) **Representative GXF Obtain.** Generate GFF3 files containing only representative transcript information.
(h) **GXF Region Overlap.** Extract the annotation information that overlaps with specific intervals based on the GFF3/GTF file.
(i) **GXF Select.** Extract annotation information related to a given ID.
(j) **Gene Density Profile.** Calculate gene density for any given genome.

17.2.2.3 Graphics

To facilitate users to perform more visual data analysis, we have developed a Java plotting engine, JIGplot, from scratch. On this basis, visualization functions of a series of bio-information data analyses are implemented.

1. **Color Palette**
   (a) **Color Picker**
   (b) **Discrete Color Scheme Generator**
2. **Heatmap Illustrator**
   (a) **HeatMap**
   (b) **Cubic HeatMap**
   (c) **Layout HeatMap**
   (d) **eFP Graph**
3. **SeqLogo**
4. **Venn and Upset Plot**
   (a) **Venn**
   (b) **Upset Plot**
5. **Basic PCA Analysis**
6. **Volcano Plot**
7. **BioSequence Structure Illustrator**
   (a) **Gene Structure View (Advanced)**
   (b) **Basic BioSequence View**
   (c) **Visualize Motif Pattern**
   (d) **Visualize Domain Pattern (Batch-CDD / Pfam)**
   (e) **Visualize Gene Structure**
   (f) **Parse MAST XML File**
8. **Show Gene on Chromosome**
   (a) **Gene Location Visualize (Advanced)**
   (b) **Gene Location Visualize from GTF/GFF**
   (c) **Map Genes on Genome from Sequence Files**
   (d) **Gene Location Visualize (Basic)**
   (e) **Circle Gene View**
   (f) **Basic Circos**
9. **Advanced Circos**
10. **Synteny Visualization**
    (a) **Genome Length Filter**. Filter small sequence fragments from a genome sequence file.
    (b) **Genome Analysis Init.** Prepare files for comparative genomic analysis
    (c) **Quick Run MCScanX Wrapper**

(d) **One Step MCScanX**. Perform MCscan Analysis in One-click
(e) **Dual Synteny Plot for MCScanX**
(f) **Text Merge for MCScanX**
(g) **Multiple Synteny Plot**
(h) **Text Transformat for Micro-Synteny View**
(i) **Multiple Micro-synteny View**
(j) **Unlimited Synteny Plot for MCScanX**
(k) **Find Gene Block Evolutionary Path by Gene Pairs**
(l) **Genome Gene Dotplot**

11. **Multiple AB1 File View**
12. **Random Item/Figure Select**
13. **Tree Annotation.** Phylogenetic tree annotation and visualization.

17.2.3 Plugin Module

Functions included in the main program TBtools are those commonly used in daily bioinformatics data analysis. There are still many other functions that are useful and fit to certain needs, although not in great demand, such as the sequence conversion from Fastq to Fasta, PubMed search result management, Excel and text format conversion, and so on.

For these functions, we have developed the plugin module in TBtools, allowing users to install corresponding plugins for the functions they need (without re-installing TBtools or installing other plugins). There are currently two main acquisition modes for plugins.

17.2.3.1 User Community

Users can download plugin files with the extension .plugin in TBtools communities. After that, from the "Others" menu select "Plugin" and then "Install Plugin" (Fig. 17.10).

Select the corresponding .plugin file from the pop-up window and install it. Certainly, "drag and drop" is also supported.

17.2.3.2 Plugin Store Online

To make plugin installation more convenient, we have developed a "Plugin Store," which deposited all the available plugins. Users can find "Plugin Store" in "Plugin" under the "Others" catalog. After launching the plugin store, users can see a list of plugins (Fig. 17.11).

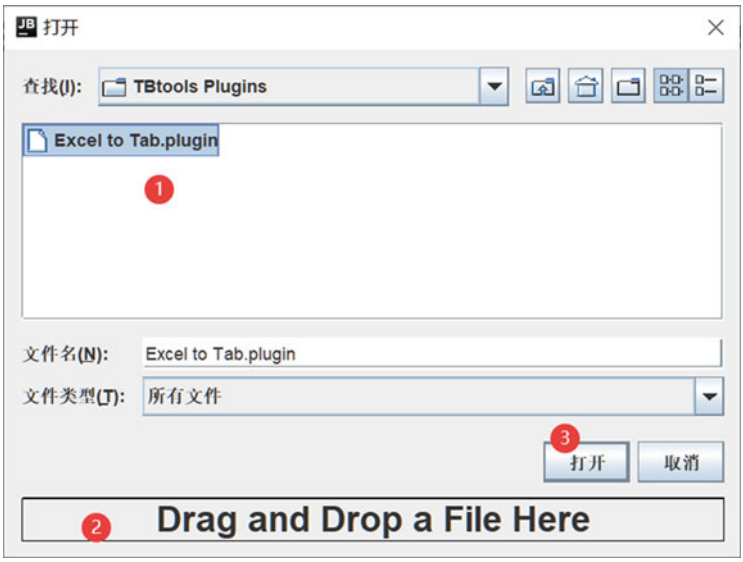

Fig. 17.10 Manually Install TBtools plugins

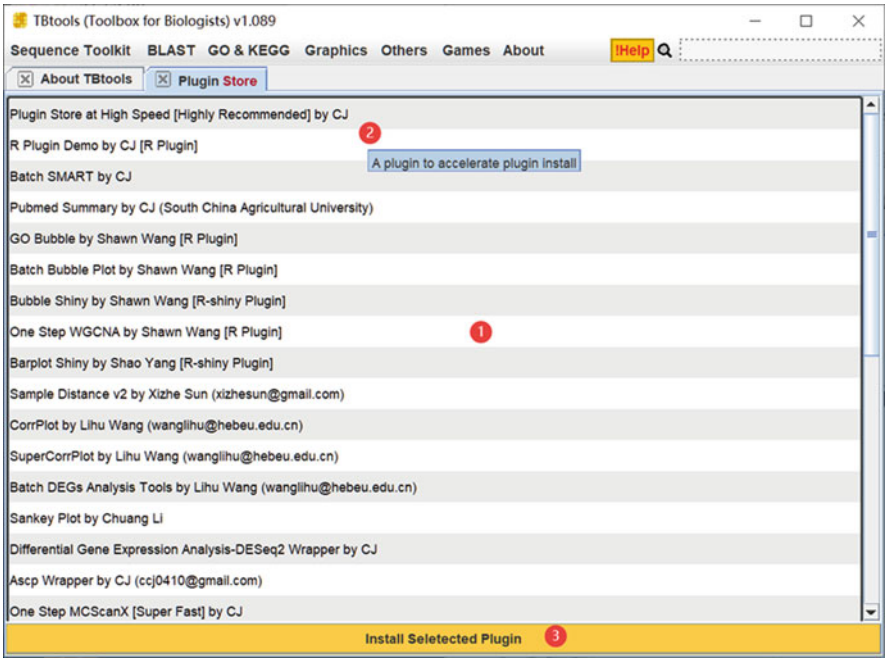

Fig. 17.11 Overview of Online Plugin Store of TBtools

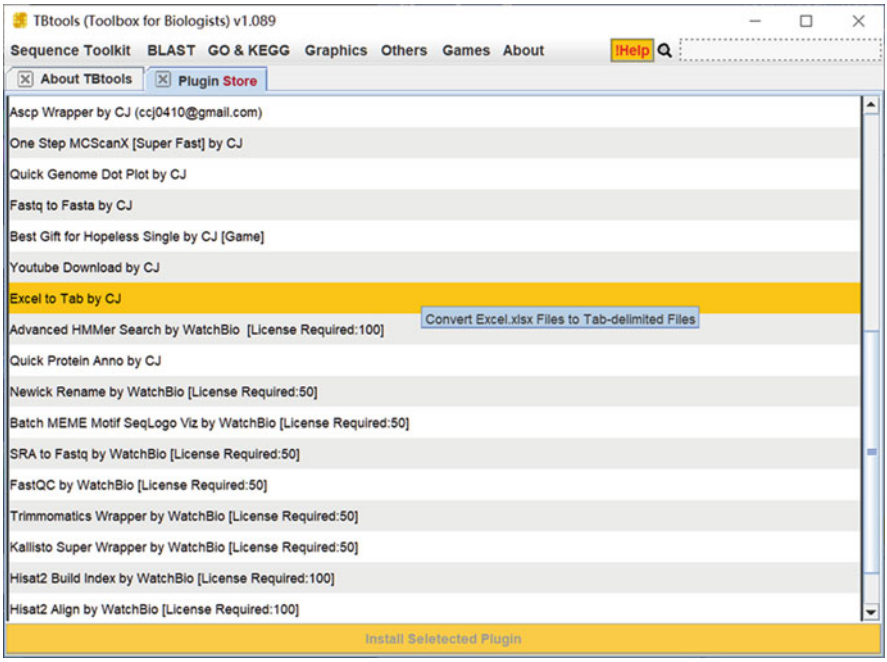

Fig. 17.12 Install a Plugin from Plugin Store

There are currently more than 30 plugins available. When the mouse hovers over a specified plugin, users can see a brief function description of the plugin. Select an ideal plugin item and click "Install Selected Plugin" to install it (Fig. 17.12).

17.2.3.3 Senior Users Participate in Plugin Development (R Plugin)

TBtools also supports users' participation in software development. Recently, we developed Rserver "plugin", an R runtime environment plugin (windows and mac version), to support direct running of R scripts. Based on this, we further developed the "R Plugin Demo" plugin. As long as users have a runnable R language script, a simple configuration can make the script into a TBtools plugin that can be used by others.

Here is an example, assuming that users currently have a script named **script.r**. Its content is

```
### Parameter acquisition
argv <- commandArgs(TRUE)
expfile <- argv[1]
title <- argv[2]
logTran <- argv[3]
colorSet <- argv[4]
titleColor <- argv[5]

### Dependent package detection and installation
if (!require('ggplot2')) install.packages('ggplot2',
repos='https://mirrors.tuna.tsinghua.edu.cn/CRAN/')
if (!require('reshape2')) install.packages('reshape2',
repos='https://mirrors.tuna.tsinghua.edu.cn/CRAN/')

### Data Processing
library(ggplot2)
library(reshape2)
expMat<-read.table(expfile,header = T,sep="\t")
head(expMat)
expMat<-melt(expMat)
if(logTran=="true") expMat$value<-log(expMat$value+1)
p<-ggplot(expMat)
p+geom_density(aes(x=value,fill=variable),alpha=I(1/4))+
  labs(title=title)+
  scale_fill_brewer(palette=colorSet)+
  theme(plot.title=element_text(size=25, hjust=0.5, face="bold", colour=titleColor,
vjust=-1))
```

The script can be invoked with the following command.

```
Rscript script.r "fpkm.xls" "R-ggplot2 BarPlot" "false" "Set1" "#e31a1c" "OutDir"
```

Users only need to prepare a few files which could be found in the "R Plugin Demo" plugin (most of them are optional) and configure the **config.txt** file to complete the plugin development (Fig. 17.13). To date, nearly ten senior users have turned their R scripts into TBtools plugins, covering a series of functions.

1. Batch Bubble Plot (desktop and shiny version)
2. Barplot (shiny version)
3. Gene co-expression Analysis (WGCNA)
4. Sample co-relation analysis
5. Differential Gene Expression Analysis (DESeq2/edgeR)
6. Sankey Plot

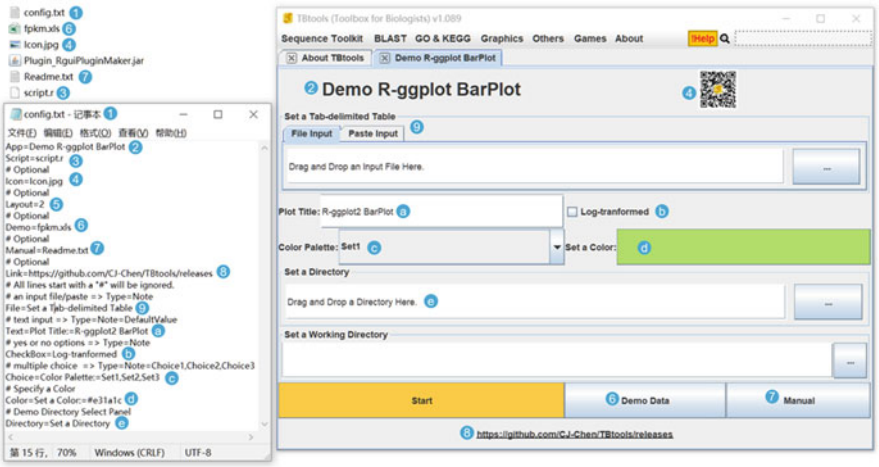

Fig. 17.13 A Demo to Make a TBtools R plugin

17.3 Demonstrations

The first step in learning swimming is getting into the water. In this section, we select four popularly used functions to demonstrate the powerful functionality of TBtools.

17.3.1 Genomic Feature Sequence Extraction Based on GFF3/GTF

With the rapid development of sequencing technologies, more and more genomes have been sequenced, which greatly promotes the scientific research on genomics. Effectively extracting and using genomic sequences becomes a routine task. TBtools provides a robust function, "GXF Sequence Extract," for quick extraction of certain sequences from genome sequences. According to the IOS logic, users only needs to set the Input and Output:

(a) **Input data.** Gene structure annotation information (GFF3/GTF format) and genome sequence (FASTA format) files of a species.
(b) **Output file.** Output file (FASTA format) path.

Most commonly, this function is used for extraction of coding sequences (CDS) or regulatory sequences (e.g., promoter).

17.3.1.1 Coding Sequence (CDS) Extraction

Open the TBtools software and select the function "Sequence Toolkit" -> "GFF3/GTF Manipulate" -> "GXF Sequence Extract" (Fig. 17.14).

1. Set the gene structure annotation information file (GFF3 or GTF format).
2. Click the "Initialize" button.
3. After that, the "Start" button will change from unavailable (gray) to available (black).
4. Select the sequence feature tag to indicate which type of sequences to be extracted. Here, it is "CDS".
5. In a GFF3 file, a sequence feature record (Feature Tag, such as CDS, EXON, mRNA) corresponds to a sequence segment of the genome and generally has grouping information with a unique ID information tag (Feature ID, such as Parent, transcript id, gene id). To obtain the complete CDS sequences of a species, we need to combine multiple CDS segment records with unified grouping information tags. Thus, we select "Parent" here.

   After the initiation is finished, then go the extraction (Fig. 17.15).

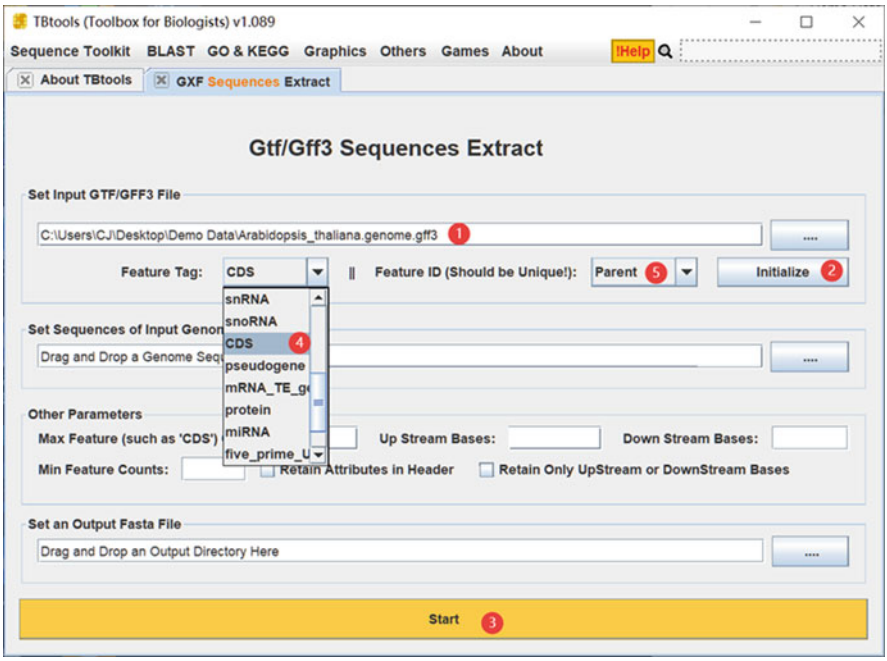

Fig. 17.14 Extraction of a Complete Set of CDS Sequences (initialization)

6. Set the genome sequence information. Please ensure that the chromosome ID is consistent between the GFF3 and the genome sequence files.
7. Set the output file path. A complete output file name is needed.
8. Click "Start" to complete the extraction. A file containing all the sequences you want will be generated in the Output directory.

17.3.1.2 Regulatory Sequence (Promoter) Extraction

In biological research, biologists usually are interested in the regulatory sequences of important genes and need to grab these sequences for further analyses. Among them, the promoter sequences of genes are the most popularly investigated. Generally, sequences of 2–3 kb upstream from the translation start site or the start codon (for these gene loci lack of UTR information) are used as a promoter sequence for analyses. TBtools can be used to get these sequences easily (Fig. 17.16).

1. Set the upstream 1000 bp sequence before CDS as the target region.
2. Check the box of "Retain only Upstream or Downstream Bases" to ensure only the specified (upstream or downstream) will be obtained; otherwise, both the upstream/downstream and CDS sequences will be extracted at the same time.
3. Set the output file path.

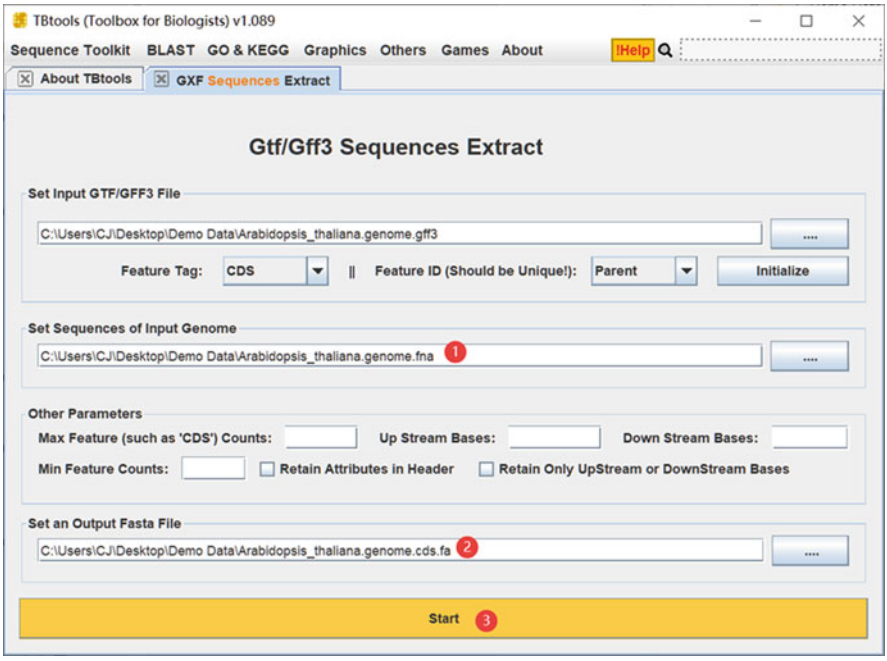

Fig. 17.15 Extraction of a Complete Set of CDS Sequences (extraction)

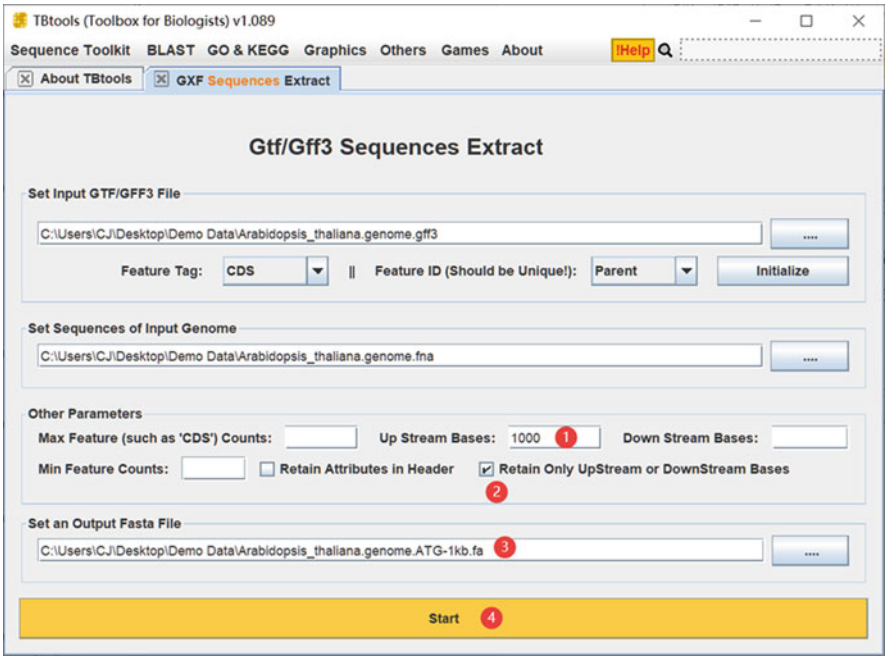

Fig. 17.16 Extraction of a Complete Set of Regulatory Sequences (promoter)

4. Click the "Start" button.

In addition, "GXF Sequence Extract" can also be used to extract a complete set of transcripts (combine multiple exon sequence together) or other sequences. The "Retain Attributes in Header" setting can be used to keep the original sequence annotation information in the output file.

17.3.2 Heatmap

Heatmap is one of the most popular graphs used for data visualization in bioinformatics data analyses. Based on its home-brew plotting engine JIGplot, TBtools provides a convenient and powerful heatmap function. Users can quickly make personalized heatmaps by using various interactive features.

17.3.2.1 Make a Simple Heatmap in a Short Time

1. Prepare a file containing a matrix of gene expression values with row and column names.
2. Paste or drag and drop the matrix file as the Input.
3. Click the "Start" button.

A heatmap plotting window will pop up instantly (Figs. 17.17 and 17.18).

17.3.2.2 Adjust the Heatmap Parameters

The heatmap graph can be personalized easily, such as data normalization, clustering of rows and columns, displaying numerical values and other information, etc. (Fig. 17.19).

1. Use the built-in color pattern to choose ideal color Scheme.
2. Use 0–1 normalization methods to format the input value matrix.
3. Cluster the data in rows and columns and display the original values in the heatmap.
4. Adjust the width of the picture (so that the number can be completely displayed in a cell).

17.3.2.3 Make a Circular Heatmap

Compared with the existing heatmap plotting tools, TBtools heatmap supports more flexible parameter adjustment. For example, users can "bend" (circularize) the heatmap to make full use of a limited space to show more information (Fig. 17.20).

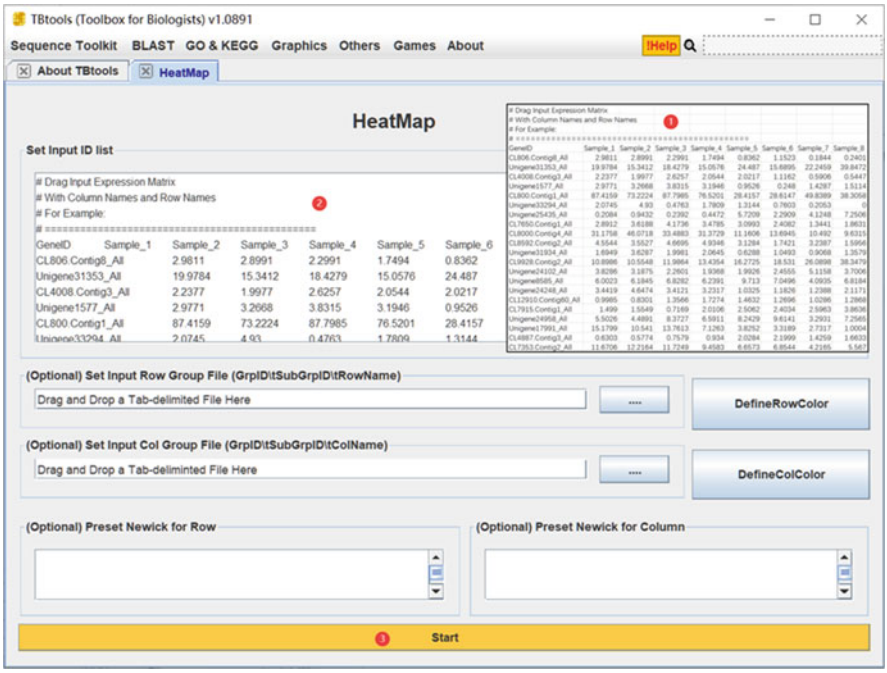

Fig. 17.17 Use TBtools to Make a Simple Heatmap

To make a circular heatmap, users only need check the "Auto Polar" box on the control panel. For further improvement, users can change the way of legend presentation by checking "Horizontal Legend".

17.3.3 Circos Plot

Circos plot is a widely used visualization approach to display large-scale genomic data. It is often used to present results at the whole genome scale to provide a comprehensive data overview. Making a Circos plot using the original package requires users to be proficient in Perl or R language programming, which inadvertently limits its application in more scientific research projects. As TBtools can easily "bend" (circularize) graphs as shown above, the Circos plotting is also supported, but in a much easier way. Users only need to prepare a few input files for graph generation without any programming or command-line operations.

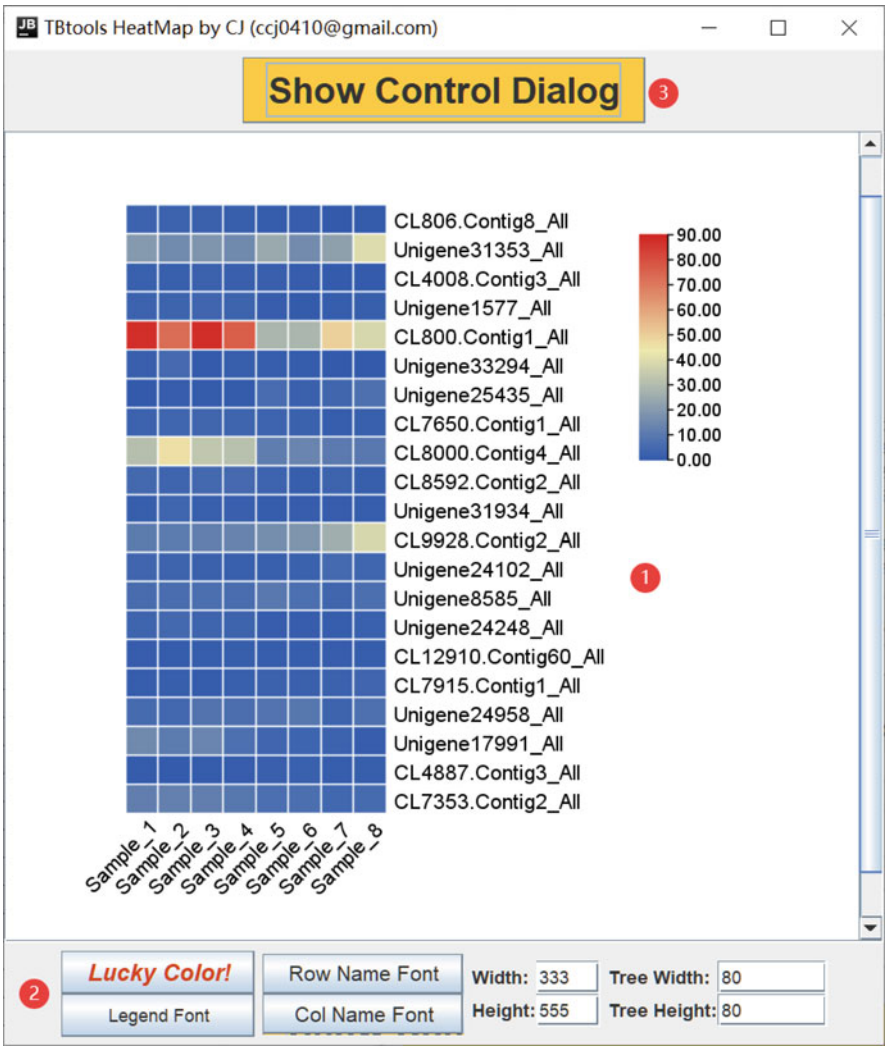

Fig. 17.18 Heatmap Plotting Window

17.3.3.1 Making the Chromosome Skeleton

With TBtools, users can make a Circos plot step by step, depending on the number of datasets to show. At first, users need to prepare the innermost chromosome skeleton track. A file containing the chromosome skeleton information ("Chromosome ID\tChromosome length" or "Chromosome ID\tChromosome starting position:Chromosome ending position") is needed, and this file can be prepared using the "Fasta Stat" function in TBtools (Fig. 17.21).

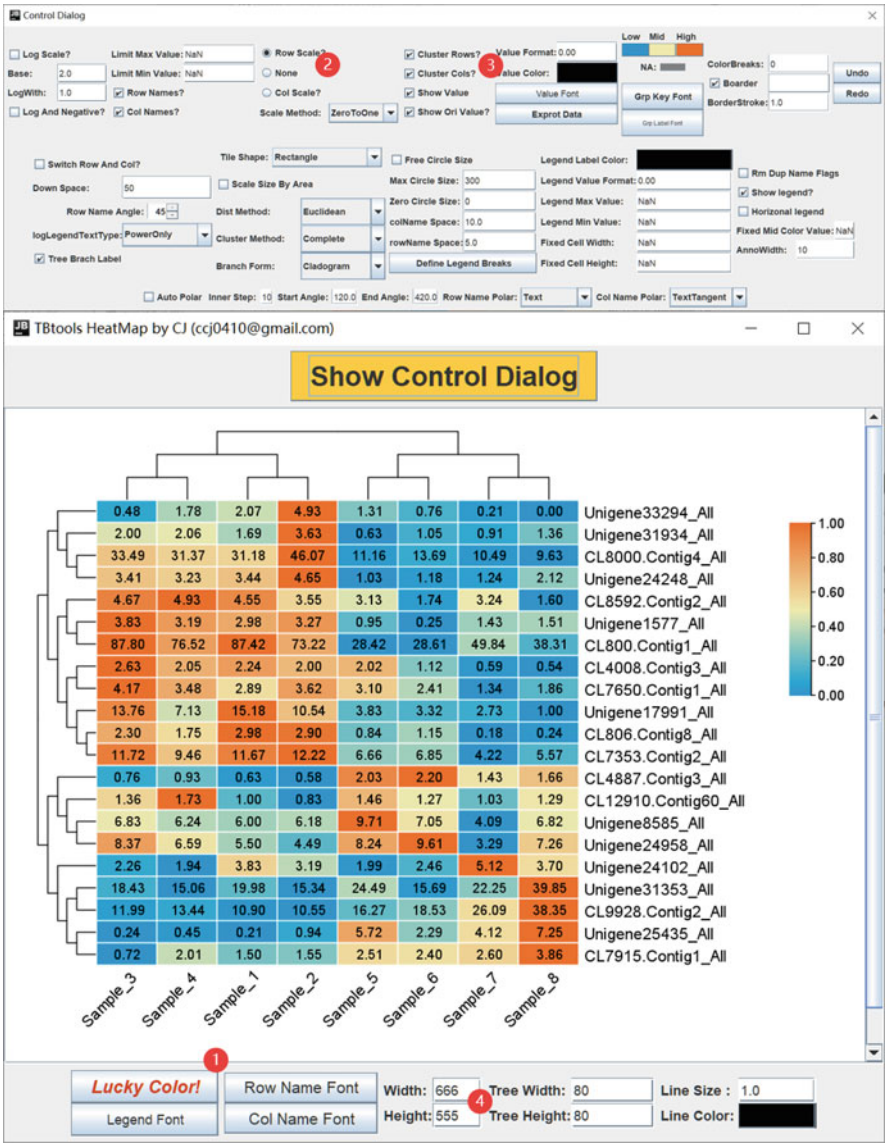

Fig. 17.19 Heatmap Parameter Adjustment (Basic)

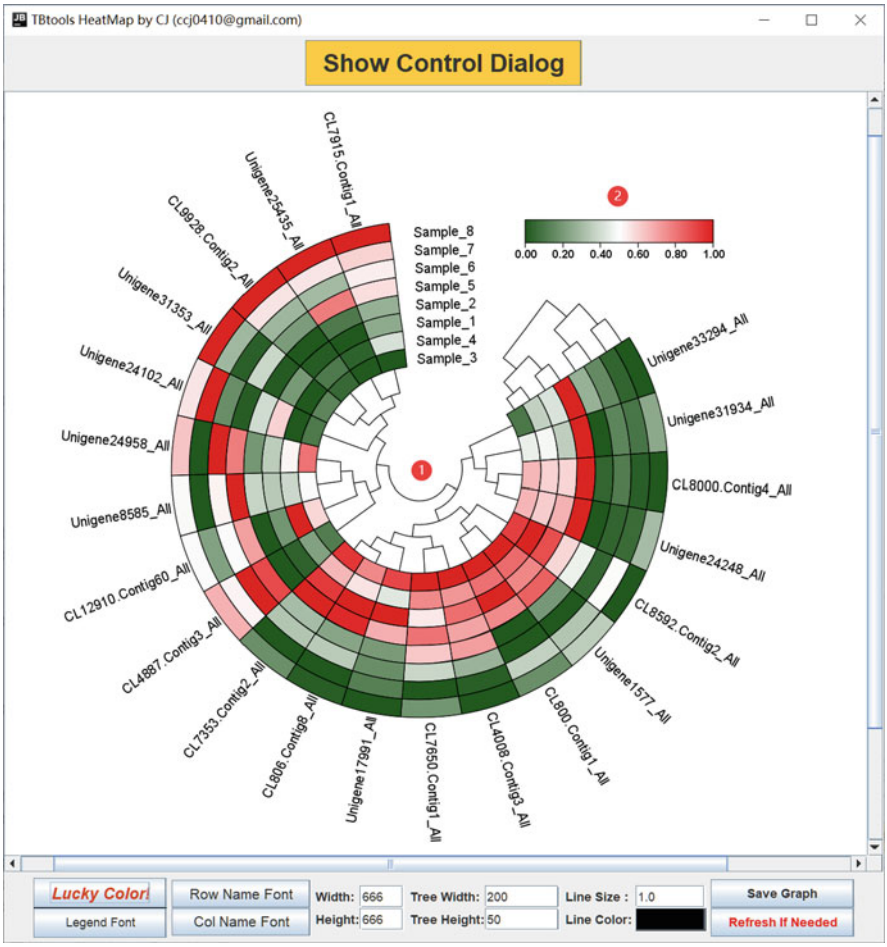

Fig. 17.20 Heatmap Parameter Adjustment (Polar Coordinates)

1. Set the genome sequence file as input.
2. Set the output file to save the length information of the chromosome sequences.
3. Check "Keep Only Sequence Length" to save only the length information to the output file.
4. Click the "Start" button.

Go to the function, "Graphics" -> "Advanced Circos", set the input file (the output file above), and "Show My Circos Plot!". Then a simple Circos graph with chromosome skeletons will be generated (Fig. 17.22).

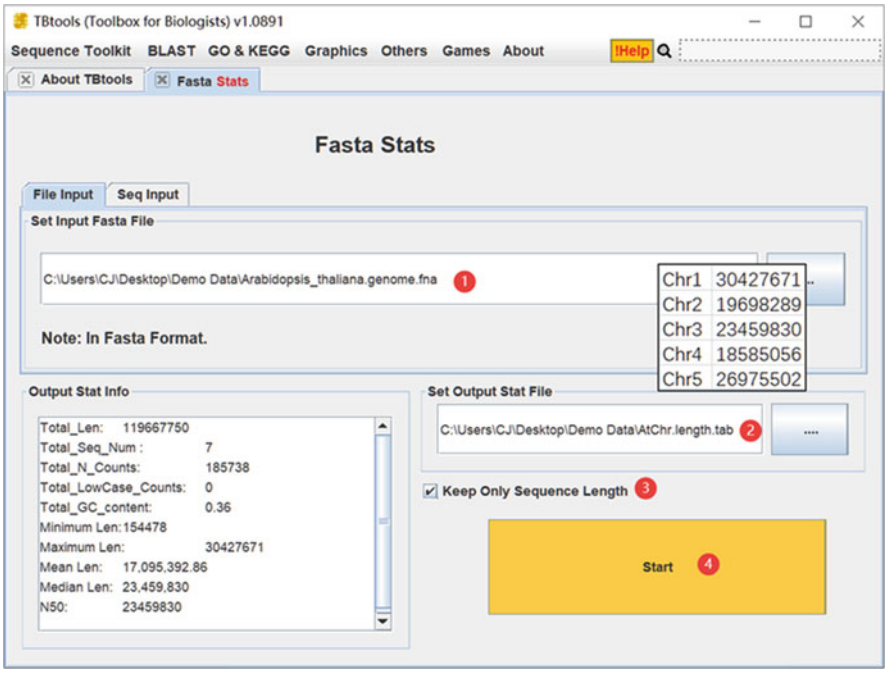

Fig. 17.21 Using "Fasta Stat" to Prepare a Chromosome Skeletons File

17.3.3.2 Display the Feature Location and Association Information of the Genome

On this simple graph, two types of information can be added: (a) Chromosome feature labels, such as the location of certain genes; (b) Chromosome segment relationships (such as large segmental duplication events). The former can be obtained through the "GXF Pos. and Info. Extract" function in TBtools; the latter can be obtained by collinearity analysis of the genome (Fig. 17.23).

1. Set the file containing information of feature locations. The format is "Chromosome ID\t Feature identifier\tChromosome starting position\tChromosome ending position\t[optional color information, R, G, B]".
2. Set the file containing interchromosomal association information. The format is "Chromosome ID\tChromosome starting position\tChromosome ending position\tChromosome ID\tChromosome starting position\tChromosome ending position\t[optional color information, R, G, B]".
3. Click "Show My Circos Plot!".
4. Users can see a Circos with more information.

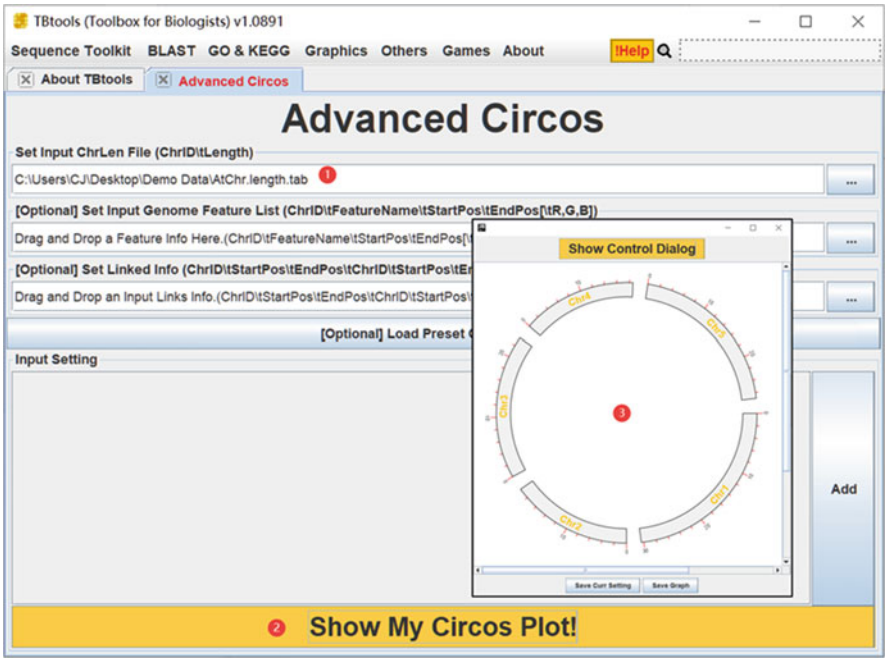

Fig. 17.22 Use "Advanced Circos" to Visualize the Chromosome Skeleton

17.3.3.3 Display Information at the Genome Scale

Usually Circos plot are used for the overview of genome-wide information, such as gene density, GC content, sequencing depth, SNP frequency, etc. These information are often recorded in a way that a chromosome region corresponds to a value. Here, we used gene density as an example, and the density information can be easily obtained using the "Gene Density Profile" function in TBtools. Open "TBtools" and select the function "Sequence Toolkit" -> "GFF3/GTF Manipulate" -> "Gene Density Profile" (Fig. 17.24).

1. Set the gene structure annotation file (GFF3/GTF) as input.
2. Set the output file path.
3. The content of the output file is formatted as "Chromosome ID\t starting position\t Chromosome ending position\tnumber of genes". As the length of each genomic interval is set to be the same, so the number of genes in each interval represents gene density.

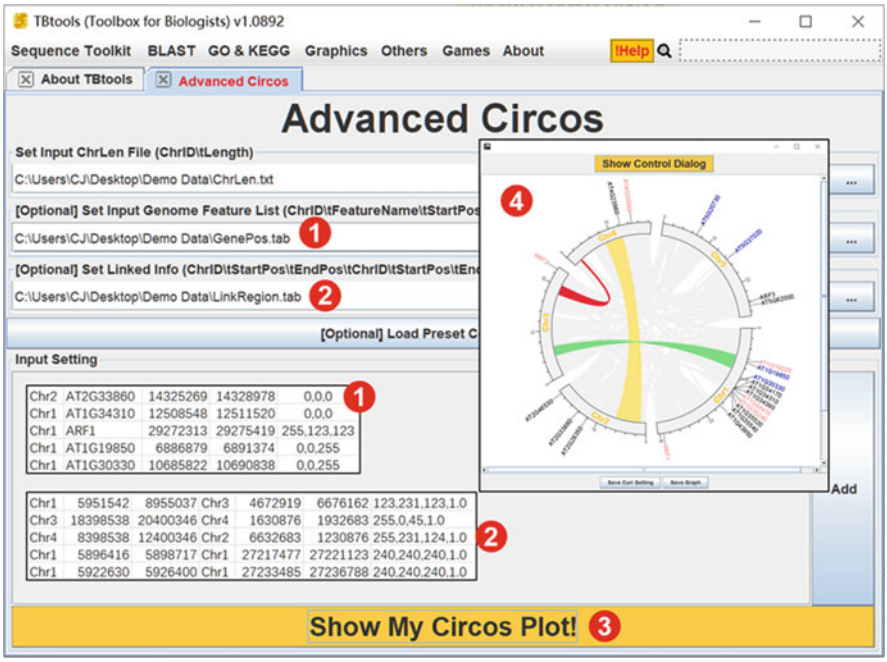

Fig. 17.23 Use "Advanced Circos" to Display Feature Locations and Collinear Regions

The resultant gene density information can be directly used for "Advanced Circos" visualization (Fig. 17.25).

1. Set the input file, which is the gene density file obtained above.
2. Adjust the Track type to "Heatmap".
3. Select the sliding window type as "None," that is, no sliding window, as the sliding window calculation has been accomplished in the "Gene Density Profile" step.
4. Gene density information is now displayed.

Data in similar formats can be displayed in different plot types. In addition to heatmap, TBtools also supports "Bar," "Line," "Point," etc. Besides, it also supports positional mark visualization, such as "Tile," "Arrow," "Triangle," etc. The input data format is "Chromosome ID\tChromosome start position\tChromosome end position\tR,G,B". Tracks of multiple data can be viewed synchronically (Fig. 17.26).

1. Overlapping of two types of tracks: "Heatmap" and "Line".
2. Use "Bar" type for the second track.
3. Support different open angles.
4. Support linear display as well.

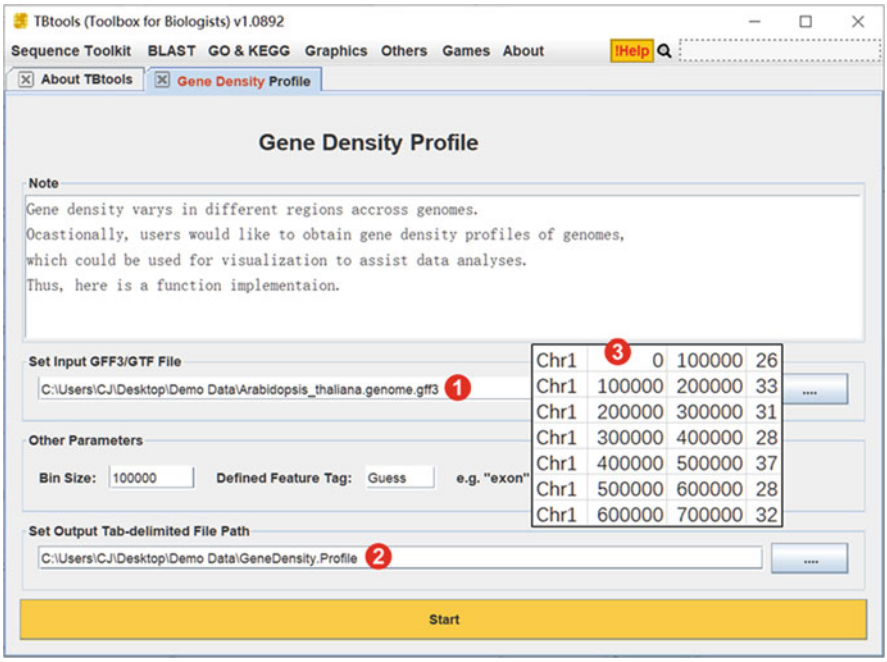

Fig. 17.24 Use "Gene Density Profile" to Get Gene Density Information

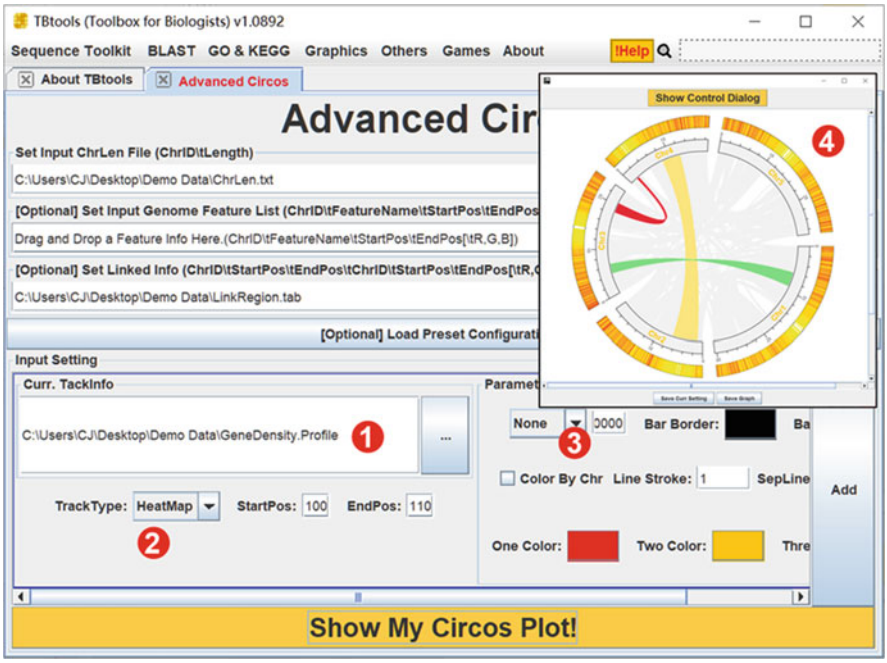

Fig. 17.25 Use "Advanced Circos" to View Gene Density over the Whole Genome

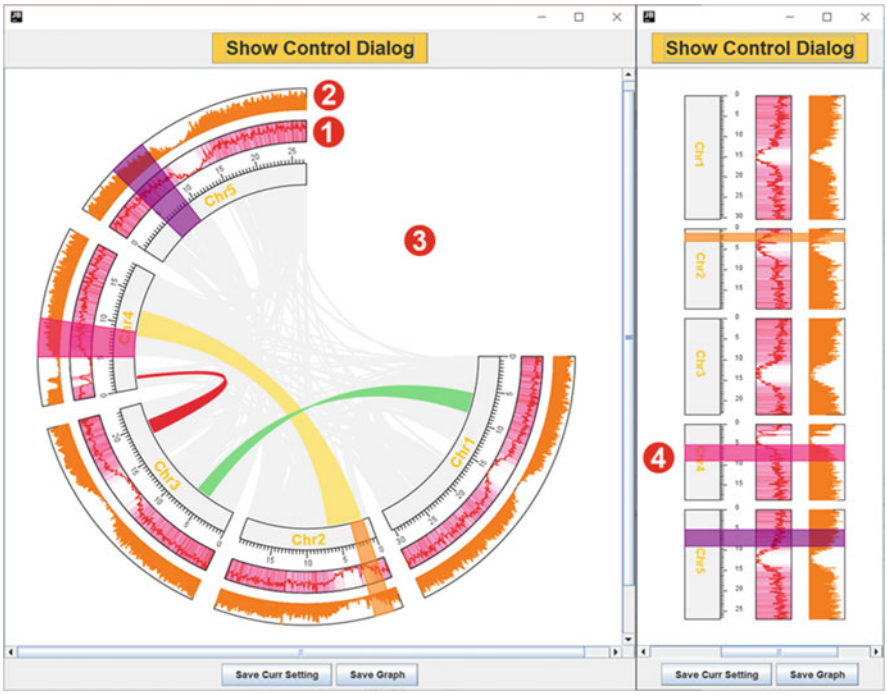

Fig. 17.26 Flexible Use of "Advanced Circos"

17.3.4 Quick Protein Functional Annotation (Plugin)

In daily data analysis, we may often obtain hundreds or thousands of new genes (unannotated), which need to be functionally annotated for biological meaning. The TBtools plugin "Quick Protein Anno" can assist users in this, which can finish the functional annotation of ~20,000 protein sequences within a few minutes and output the results into a table for further exploration. Users can install it through the plugin store. Turn to the plugin store through "Others" -> "Plugin" -> "Plugin Store".

After the installation is complete, you can run the function through the menu "Others" -> "Plugin" -> "Quick Protein Anno" (Figs. 17.27 and 17.28).

1. Set the database used for annotation. Generally, the "Swissprot" protein sequence library is used.
2. Users can click "DB Download" to download the "Swissprot" protein sequence library.
3. Set the protein sequence file to be annotated or paste the sequences directly.
4. Set an output file path.
5. The output result file formatted as "Gene ID\tHigh-frequency keyword #1\tHigh-frequency keyword #2\tOptimal comparison results".

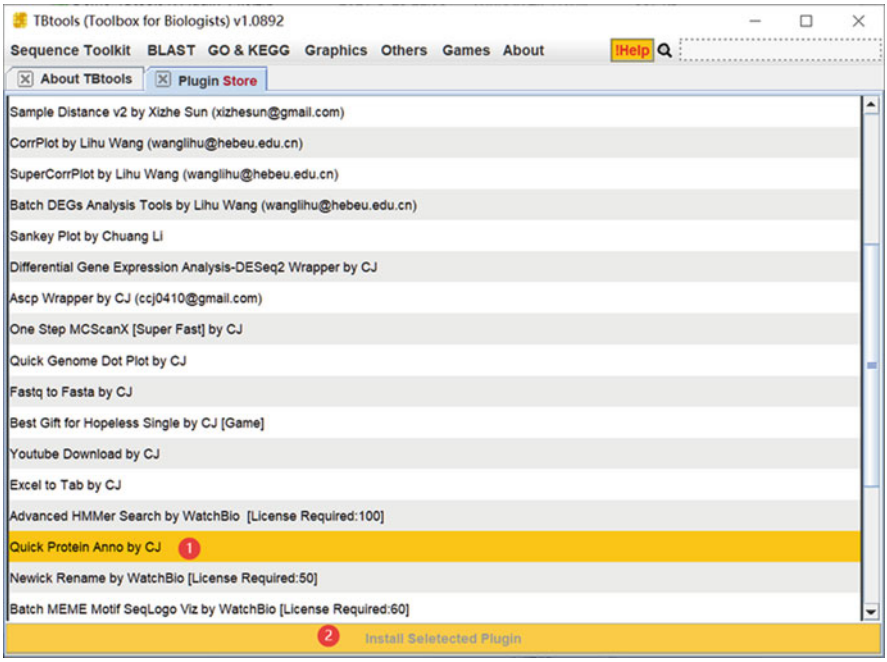

Fig. 17.27 Install the "Quick Protein Anno" Plugin through the Plugin Store

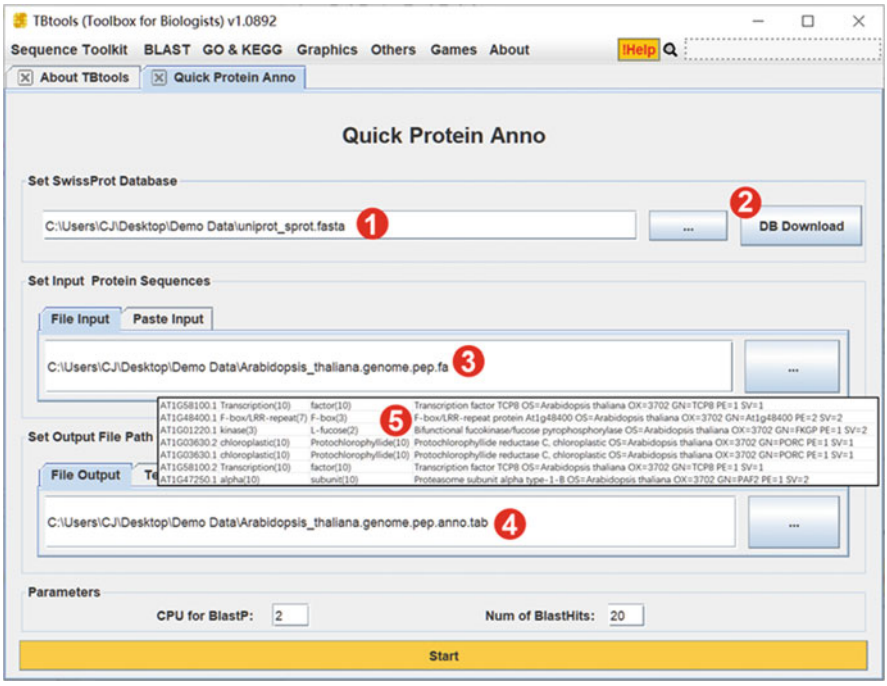

Fig. 17.28 Use "Quick Protein Anno" for Function Annotation

References

Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ (1997) Gapped BLAST and PSI-BLAST : a new generation of protein database search programs. Nucleic Acids Res 25:3389–3402

Chen C, Chen H, Zhang Y, Thomas HR, Frank MH, He Y, Xia R (2020) TBtools: an integrative toolkit developed for interactive analyses of big biological data. Mol Plant 13:1194–1202

Connors J, Krzywinski M, Schein J, Gascoyne R, Horsman D, Jones SJ, Marra MA (2009) Circos : an information aesthetic for comparative genomics. Genome Res 19:1639–1645

Wang Y, Tang H, Debarry JD, Tan X, Li J, Wang X, Lee TH, Jin H, Marler B, Guo H et al (2012) MCScanX: a toolkit for detection and evolutionary analysis of gene synteny and collinearity. Nucleic Acids Res 40:e49

Winter D, Vinegar B, Nahal H, Ammar R, Wilson GV, Provart NJ (2007) An 'electronic fluorescent pictograph' browser for exploring and analyzing large-scale biological data sets. PLoS One 2:e718