The SciRoKo 2.1 Manual

1. General Information

SciRoKo is free for use. If you intend to use SciRoKo commercially or you suffer from a surplus budget, a small donation of e.g.: 100$ or 100€ will not be rejected. All donated funds will be used for surviving a monetary ungrateful PhD-thesis or buying new hardware. Please use the following account:
IBAN: AT733439000000012427
BIC (SWIFT): RZOOAT2L390

SciRoKo is entirely written in C# and primarily designed for the use with Windows. Due to the tireless efforts of the people from the Mono-Project, SciRoKo also works in Linux, MacOS X, Free BSD, UNIX, and Solaris. See the Mono-Project: http://www.mono-project.com/Main_Page

2. Installation

1. Download the SciRoKo.zip from www.kofler.or.at/Bioinformatics/
   The SciRoKo executable is platform independent and can be used with all operating systems.
2. Unpack the archive in a folder of your choice
3. Make sure that the Microsoft .net Framework 2.0 is installed, if not ask your computer administrator or download it yourself from the vendors homepage.
   Alternatively the mono-project can be used, this approach is necessary for operating systems other than Microsoft Windows.
4. Just double click on SciRoKo.exe

Links:
-Mono-Project software: http://www.mono-project.com/Downloads
-Microsoft .net Framework 2.0:
http://www.microsoft.com/downloads/details.aspx?FamilyID=0856EACB-4362-4B0D-8EDD-AAB15C5E04F5&displaylang=en
-General information about .net and C#: http://en.wikipedia.org/wiki/.net_framework

3. Introduction

Several tools for the analysis of microsatellites already exist. (See Table 1)

Table 1. Common programs and tools for SSR search: summary of the features and properties

search tool	perfect search	mismatched search	programming language	operating system	user interface	SSR-statistics
MISA	yes	no	Perl	Unix/most os	console	yes
SSRFinder	yes	no	C	most os	console	no
SSRIT	yes	no	Perl	Unix/most os	Web interface	no
TRF	no	yes	?(C++)	most os	Windows form/Web	no
TROLL	yes	no	C++	Linux	console	no
Sputnik	no	yes	C	most os	console	no
Modified Sputnik I	yes	yes	C	most os	console	no
Modified Sputnik II	yes	yes	C	most os	console	no
SciRoKo	yes	yes	C#	Windows/most os	Windows forms	yes

In our opinion this tools have the following drawbacks: (i) except SSRIT and TRF this tools are not user friendly because they require console input or Unix/Linux as operating systems;
(ii) many of them allow only search for perfect SSRs, although an imperfection within an SSR does not necessarily mean the end of this SSR. It may continue beyond this imperfection, moreover, the imperfection may be caused by sequencing error;
(iii) Sputnik, Modified Sputnik I-II and TRF are very slow ;
(iv) Sputnik, Modified Sputnik I-II do not allow SSR search for hexanucleotide motifs (TRF only with difficulties); (v) TRF allows parameter adjustment only within a limited range and produces a cornucopia of inconvenient output files;
(vi) Finally, none of the listed tools allows detailed subsequent statistical analysis of the search results.

3.1 How the SciRoKo SSR-search algorithm works in detail:

The SciRoKo SSR-search module offers four search modes, two for perfect and two for mismatched SSR search. In the two perfect SSR search modes each nucleotide at position i is tested for identity with the nucleotide at position i+t, where t is the motif length (1-6). Upon identity i is increased i=i+1 until no further identity can be found. If this SSR meets the minimum requirements as specified in ``Minimum repeats'' or ``Minimum total length'', depending on which one of the two perfect search modes is used, the microsatellite is directly reported into the output file. The score of perfect microsatellites equals their total length.

In the two mismatched SSR search modes, perfect SSRs (SSR-seeds) act as origin for subsequent 5' and 3' extensions. The minimum requirements for the SSR-seeds can be set as low as 2 repeats or 3 nucleotides. Whether a microsatellite is finally reported into the output file depends only on the achieved score. The SciRoKo scores are calculated according to the first two equations:

1. Score_Fixed_Penalty = hits – mmP * mm
2. Score_Variable_Penalty = hits – mm * (m_L * mmP)
3. Score_Sputnki = hits - m_L – mmP * mm

The parameters are: hits (matches with the virtual perfect microsatellite; see below), number of mismatches (mm), mismatch penalty (mmP) and the length of the SSR motif (m_L). Equation 1 is used in the ``Fixed mismatch penalty'' mode, equation 2 in the ``Variable mismatch penalty'' mode and equation 3 is used in the Sputnik family SSR search tools.
If the score of an imperfect SSR achieves the ``Required score'', the SSR is reported into the output file.
The process of SSR-extension used in the two mismatched SSR search modes progresses in major loops and mismatch permutations (See Figure 1)
Initially the SSR-seed is set as highscore SSR. The highscore SSR acts as origin for the 5' and 3' major loops. Within a major loop a number of mismatch permutations (see below) is

created. The mismatch permutation achieving the highest score is called permutation highscore. If the permutation highscore is equal or higher than the highscore, the permutation highscore is set as the new highscore and acts as origin of the next major loop. The SSR-seed is at first 5' extended with major loops until the permutation highscore is lower than the highscore and then 3'. Three types of mismatches can be found in an SSR: deletions, base substitutions and insertions. Within a major loop a recursion creates for a given ``Max mismatches at once" (mmao) all possible combinations of mismatches, allowing for perfect microsatellite stretches between the mismatches: $3^{mmao}+3^{mmao-1}+\ldots+3^{1}$
The recursion is aborted premature if the end of the file or the end of a previous SSR has been reached. Branchings within the recursion only occur at mismatch sites. For the mismatched SSR-search SciRoKo creates a virtual perfect microsatellite (vpm) from the SSR-seed motif, starting at the first position of the SSR-seed (Figure 1). The vpm continues indefinitely in the 5' and 3' direction and acts as template for comparisons with the DNA sequence. Initially the position pointer moves one position from the SSR-start or SSR-end to the 5' or 3' direction for 5' or 3' extension respectively.
Subsequently, the position pointer compares each nucleotide in the DNA sequence with the corresponding nucleotide in the vpm. Each of the three mismatch types has an own distinct pattern (Figure 1). In a major loop all possible mismatch permutations, i.e. combination of mismatch patterns, are tested and the mismatch permutation achieving the highest score is set as the permutation highscore.

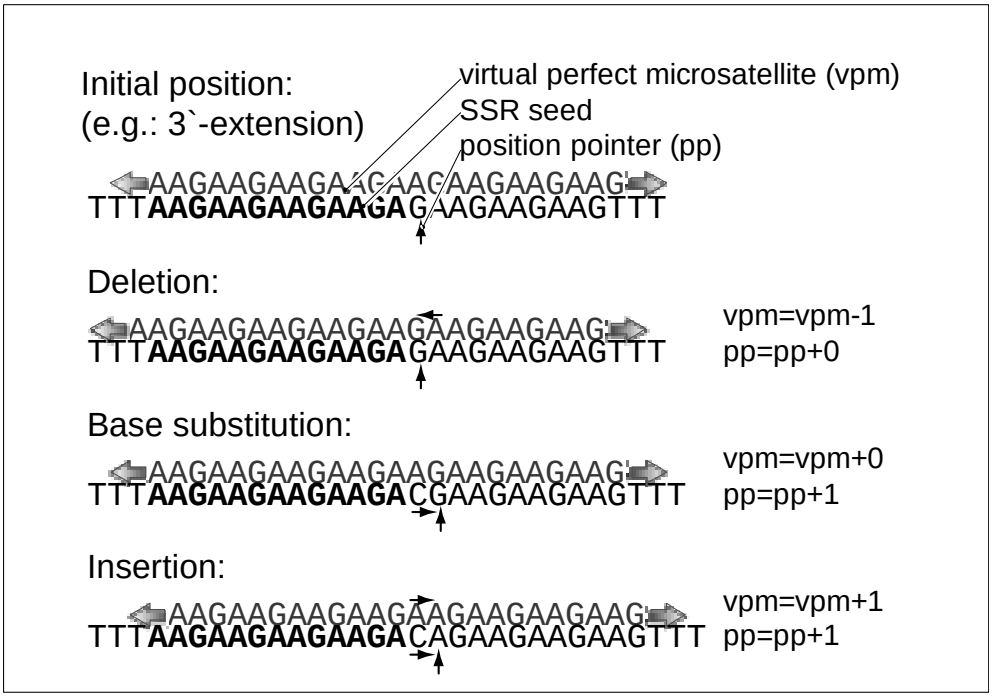


Fig. 1: Pattern of mismatches at a recursion branching site during 3'-extension. For the identification of a deletion the virtual perfect microsatellite (vpm) is moved one bp to the 5'-direction. For a base substitution the position pointer is moved one bp to the 3'-direction and for a insertion the vpm and the position pointer are moved one bp to the 3'-direction.

4. The SciRoKo SSR-search Module

The SciRoKo main menu. To perform SSR search follow the steps in numerical order.

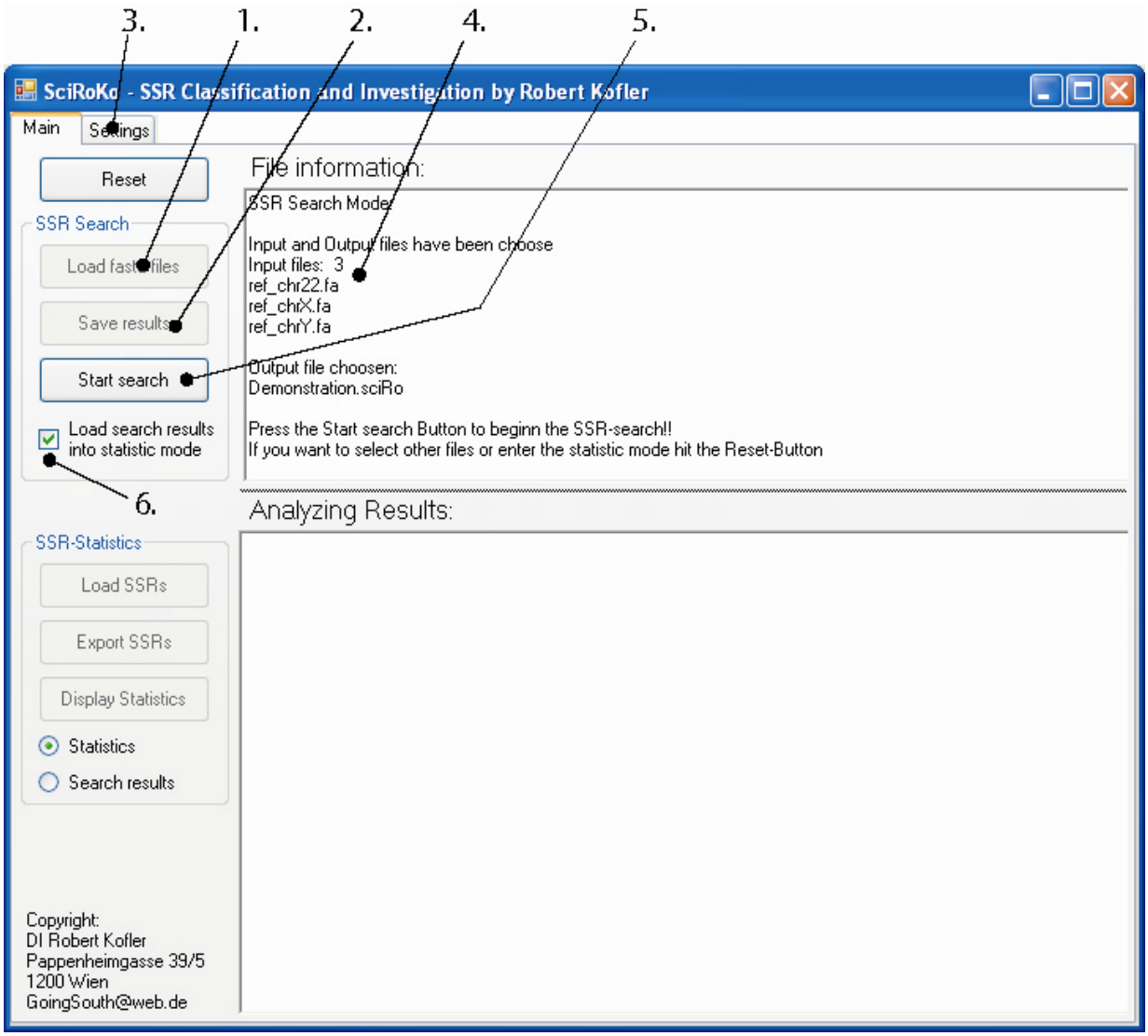
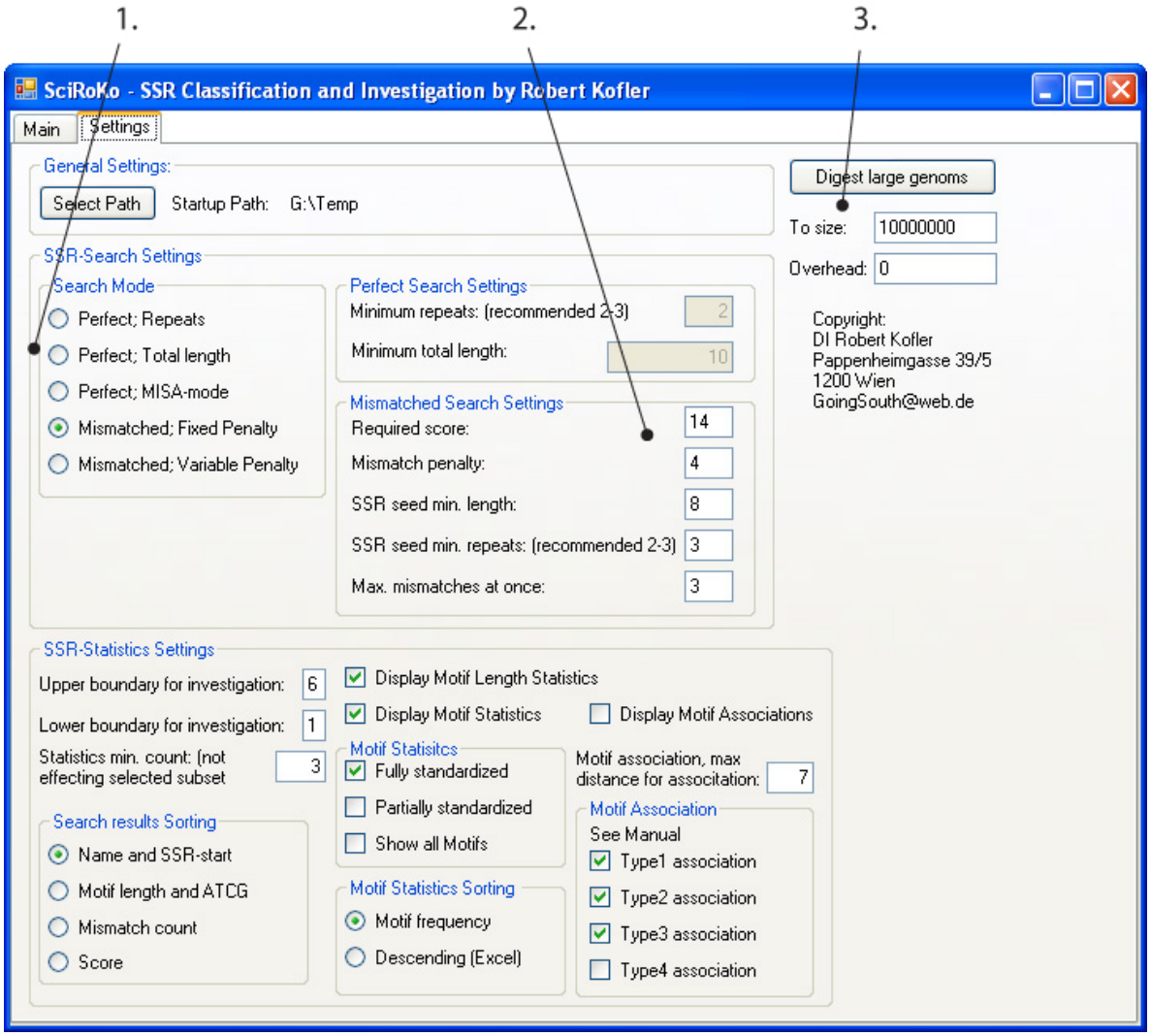


Fig. 2.: SciRoKo 2.1 main menu

1. Choose an input file. SciRoKo accepts all fasta files with the extensions *.fa ; *.fasta; *.txt ;
   A single fasta file might contain multiple fasta sequences separated through the character '>'. Additionaly SciRoKo accepts multiple fasta files at once. For instance, the whole rice genome can be investigated at once, with each chromosome representing a single file or with all chromosomes copied into a single file.
2. Choose the output file for the SSR search. Two file types are supported as output formats. The SciRoKo format (*.sciRo) and the Tab delimited format (*.td). If exporting of the SSR-search results into the sputnik-family file formats is required, the SSR-search results have to be loaded into the SSR-statistics module prior to exporting (See Chapter 5).
3. Adjust the settings used for SSR-search (see below)
4. Make sure the appropriate files have been chosen for input and output. Press the Reset-Button to choose different files.

5. Press the Start button to start SSR search
6. When this box is checked the SSR-search results are directly loaded into the SSR-statistics module. Even when checked, the SSR-search results are first reported into the chosen output file.

4.1 Adjust the SSR-search settings:



1. First choose the SSR search mode. SciRoKo offers three modes for perfect SSR search, one for SSR search according to total length and two according to the total repeats. The “Perfect; MiSA-mode” requires an input string of the form *mono-di-tri-tetra-penta-hexa.* For instance, the input string 12-6-7-5-5-5 states that a trinucleotide microsatellite has to have at least 7 repeats.
   Additionally SciRoKo provides two mismatched SSR search modes, one using a fixed mismatch penalty and one using a variable mismatch penalty.
2. Adjust the settings used for SSR search such as total length or required repeat number in the perfect SSR search modes.
   When using the mismatched SSR search modes adjust the mismatch penalty, the required score, the requirements for the SSR-seed and the maximum number of mismatches allowed in a row (max. mismatches at once is equivalent to the depth of the recursion). An SSR-seed is each perfect microsatellite meeting the specified

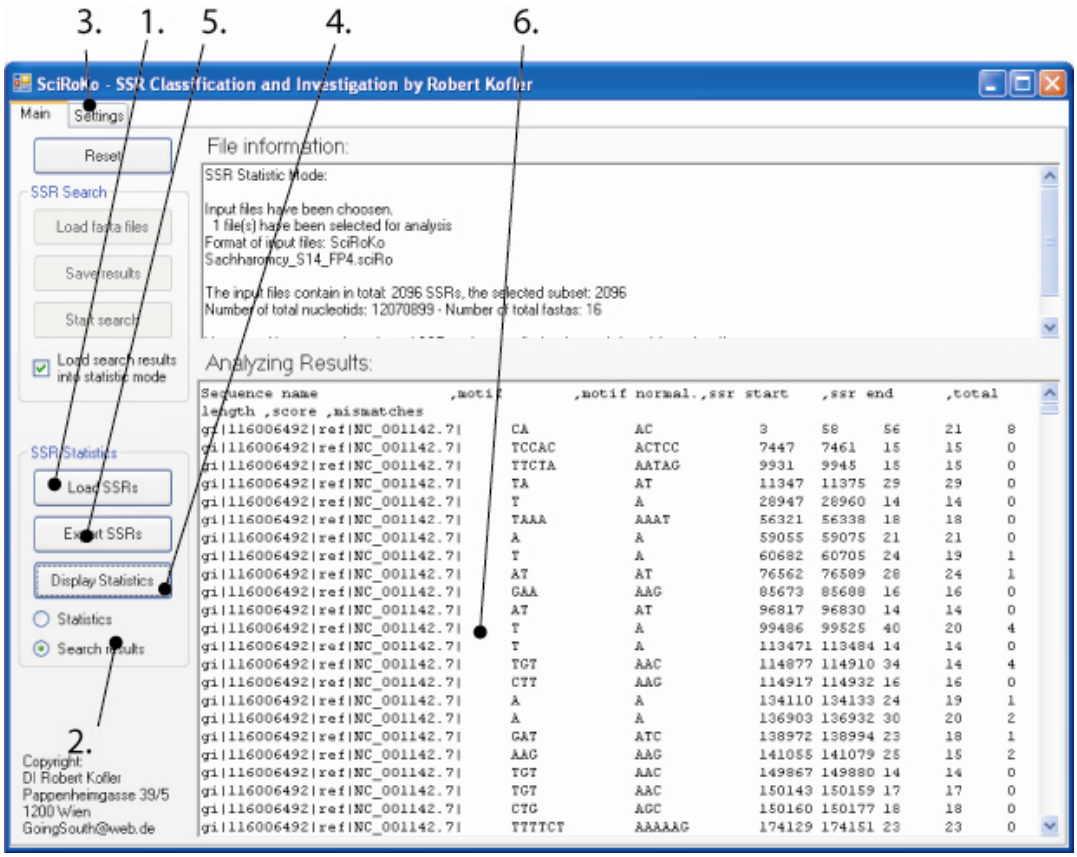
requirements like minimum repeats or minimum total length. The lower boundary for the SSR-seed settings is: 2 repeats and a minimum length of 3.

3. Large genomes, like the human, have chromosomes larger than 200 Mbp. Unfortunately 200 Mbp are to large for ad hoc analysis with SciRoKo 2.1, SciRoKo accepts fasta files with a size up to 50 Mbp. It is therefore necessary to digest large genomes into smaller chunks of the chosen size. We recommend using a chunk size of 50 Mbp with no overhead specified.
   Once pre-treated SciRoKo analysis the whole human genome in 460 seconds.

   The pre-treated chromosome chunk files are stored in a subfolder, file names and file number are kept identical.

5. The SciRoKo SSR-Statistics Module

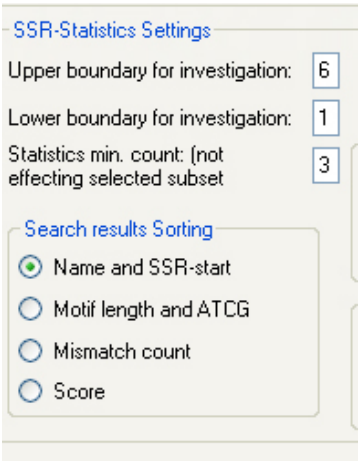
5.1 Display the SSR-search results:



1. Choose the input files for the SSR-statistics module. Although multiple input files can be selected it is not recommended, because this might cause problems with the total number of nucleotides or files
   Important note: Adjust the file-extensions of the Sputnik, Modified Sputnik I-II files prior to use with the SciRoKo SSR-statistics module. The following list exhibits the required file extensions.

Sputnik	*.sput
Modified Sputnik I	*.m1sput
Modified Sputnik II	*.m2sput

2. Select the display search results radio button
3. Adjust the settings:



A lower and an upper boundary might be specified. Only microsatellites within this boundary are displayed. For example lower boundary = 2 -> dinucleotide SSRs; upper boundary = 5 -> pentanucleotide SSRs.

a. The SSR search results might be sorted according the name of the fasta sequence name and with equal sequence name according the SSR-start position. This sorting is also used for within the compound microsatellite identification algorithm;

b. The found microsatellites can be sorted according the motif length and with equal motif length in descending valuation ATCG. This sorting is used in the two standardization MotifMatrices (see below). Descending valuation ATCG means that motif variations containing the most A are displayed first then the motif variations with the most T and so on.

c. The microsatellites could also be sorted according the number of mismatches. For instance, with this feature it is easily possible to identify the microsatellites with the most mismatches in the whole human genome.

d. The SSRs can also be sorted according their score, allowing identification of the "highscore" microsatellites.

4. Hit the Display statistics button to display the search results
5. The subset of the microsatellites (within the specified parameters) might also be exported using the selected sorting into the SciRoKo, Sputnik, Modified Sputnik I-II and Tab delimited file formats.
6. Display; Exhibits the SSR-search results.

5.2 Microsatellite statistics:

SciRoKo generates three different statistic outputs; Motif length infos, Motif infos, and Motif association statistics (compound microsatellites).

To allow categorization and statistical analysis of the identified microsatellites SciRoKo standardizes the identified microsatellites in two intensities: **Full and partial standardization**.

The Motif association statistic requires the full spectrum of standardizations. The Motif length info only requires standardization according to the motif lengths (e.g.: all trinucleotid microsatellites are grouped into the same category).

During the standardization process similar microsatellite motifs are grouped together. For instance the microatellite motifs “AG” and “GA” become identical during the process of partial standardization yielding the partially standardized motif “AG”.
During full standardization also the reverse complements of microsatellite motifs are considered. For instance full standardization groups the microsatellite motifs “TC”, “CT”, “AG”, and “GA” together into one group (“AG”).
To facilitate the standardization process SciRoKo contains two hardcoded MotifMatrices which contain each possible microsatellite motifs. Microsatellite motifs are searched in the Motif Matrices and the standardizations are returned.

A	*T*				
C	*G*				
AT	*TA*				
AC	*CA*	*GT*	*TG*		
AG	*GA*	*CT*	*TC*		
CG	*GC*				
AAT	*ATA*	*TAA*	*ATT*	*TTA*	*TAT*

Fig 3.: Excerpt from the Motif Matrix, fully standardized. Related motifs are arranged in one row. The left motif represents the fully standardized motif.

The MotifMatirx is sorted according to two rules: (i) short motifs are displayed first (mononucleotide motifs first, followed by the dinucleotide motifs and so on) and (ii) motifs are arranged with descending valuation A-T-C-G (Remember: the motifs with the most A first than the T and so on)
This arrangement ensures a high speed for the standardization process since mononucleotide and dinucleotide motifs are extremely abundant. Therefore an early standardization of the most abundant motifs saves a lot of computation time.
Palindromic microsatellite motifs have also been considered during construction of the MotifMatices (e.g.: CG or ACGT)

A		
T		
C		
G		
AT	*TA*	
AC	*CA*	
AG	*GA*	
TC	*CT*	
TG	*GT*	
CG	*GC*	
AAT	*ATA*	*TAA*

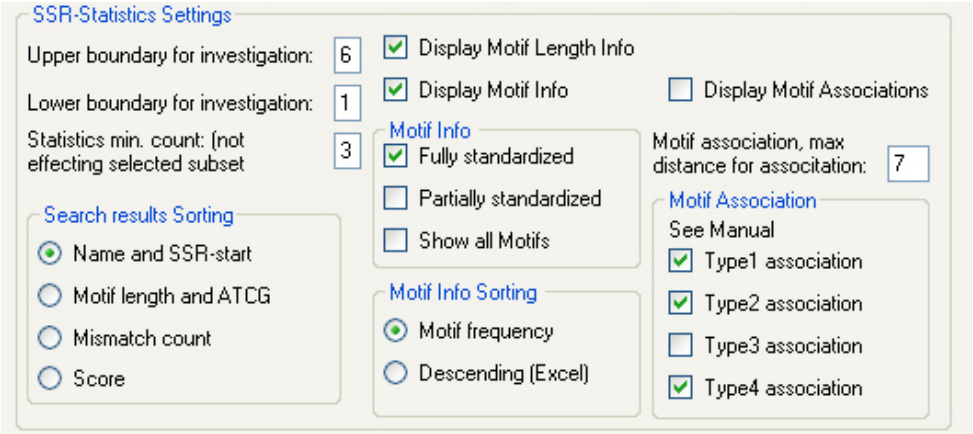
Fig. 4: Excerpt from the Motif Matrix, partially standardized. The left motifs represent the partially standardized variations.

.

Display microsatellite statistics:

1. First choose the Statistics Radio Button in the main menu

2. Then choose the settings: Which type of the statistics should be displayed? Display the motif length infos? Display the motif infos? Display the motif association? A frequency threshold might be specified (statistics min. count). This feature is especially handy for the motif association statistics.



3. Press the Display statistics button in the main menu

5.2.1 Motif Length Statistics (mono-, di-, tri- etc nucleotide SSRs)

Motif	Counts	Av.length	Av.MisM.	Nucleotides/SSR	Files/SSR	Counts/Mbp	GC Content
mononucleotide	618	20,43	0,54	19532,20	0,03	51,20	0,02
dinucleotide	367	29,04	1,54	32890,73	0,04	30,40	0,18
trinucleotide	628	26,93	1,27	19221,18	0,03	52,03	0,33
tetranucleotide	141	18,92	0,40	85609,21	0,11	11,68	0,16
pentanucleotide	137	21,58	0,69	88108,75	0,12	11,35	0,29
hexanucleotide	205	29,82	1,12	58882,43	0,08	16,98	0,43

Fig. 5: Excerpt from the Motif Length statistics of Saccharomyzes cereviseae. Detailed information for the frequencies, average mismatches etc are displayed for each motif length category.

The Motif Length Statistics calculate comprehensive statistic information for mono-, di-, tri-, tetra-, penta- und hexanucleotide microsatellites:

Column 1: Motif length category: mononucleotide, dinucleotide etc

Column 2: the total counts
Column 3: the average length of a microsatellite from this length category
Column 4: the average number of mismatches
Column 5: number of nucleotides / microsatellite; e.g.: one mononucleotide SSR each 19500 nucleotides
Column 6: the number of files / microsatellite; this feature is only importand for enriched librarys or BAC end sequences etc
Column 7: the average counts per million base pairs of a SSR from this length category
Column 8: the average GC content. For the average GC content the whole SSR-sequences including the mismatches are considered. Thats why a pure AT-microsatellite can achieve a GC-content of 0,02 = 2%

Also a subset can be chosen, for example hexanucleotide microsatellites can be excluded for a better comparision with Sputnik results.

5.2.2 Motif statistics:

Related microsatellite motifs are grouped together and common group specific features are computed. Motif statistics contain two subcategories, fully and partially standardized motif statistics.

:

Motif	Counts	Av.length		Av.MisM.	Nucleotides/SSR	Files/SSR	Counts/Mbp	GC Content
A	614	20,46	0,54	19659,44	0,03	50,87	0,02	
AT	307	24,65	0,79	39318,89	0,05	25,43	0,02	
AAG	153	23,84	1,06	78894,76	0,10	12,68	0,34	

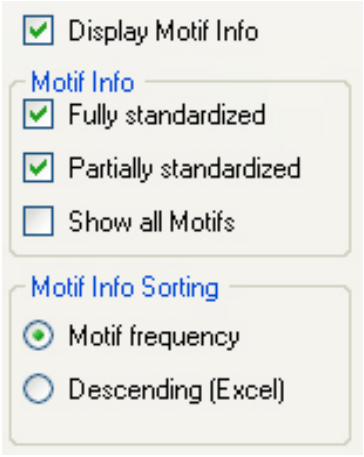
Fig. 6: Excerpt from fully standardized motif statistics (Saccharomyzes cerevisaea)

Motif	Counts	Av.length		Av.MisM.	Nucleotides/SSR	Files/SSR	Counts/Mbp	GC Content
A	319	20,10	0,51	37839,81	0,05	26,43	0,02	
AT	307	24,65	0,79	39318,89	0,05	25,43	0,02	
T	295	20,85	0,58	40918,30	0,05	24,44	0,02	
TTC	77	26,01	1,34	156764,92	0,21	6,38	0,34	
AAG	76	21,63	0,78	158827,62	0,21	6,30	0,34	

Fig. 7: Excerpt from partially standardized motif statistics (Saccharomyzes cerevisaea)

The columns of the motif statistics are similar to the motif length statistics:
Column 1: standardized motif (fully or partially)
Column 2: total counts
Column 3: the average length of a microsatellite from this fully or partially standardized m.
Column 4: the average number of mismatches
Column 5: number of nucleotides / microsatellite; e.g.: one poly-A SSR each 19659 nucleotides
Column 6: the number of files / microsatellite; this feature is only importand for enriched librarys or BAC end sequences etc
Column 7: the average counts per million base pairs of a SSR from category
Column 8: the average GC content. For the average GC content the whole SSR-sequences including the mismatches are considered. Thats why a pure AT-microsatellite can achieve a GC-content of 0,02 = 2%

Settings for motif statistics:



The motif statistics can be sorted in two ways.

a. According the total count (Column 2)
b. According the motif length (mononucleotide motifs first, dinucleotide motifs second etc) and with equal motif length in descending valuation A-T-C-G. This feature facilitates comparison of the SSR-search results with Microsoft Excel.

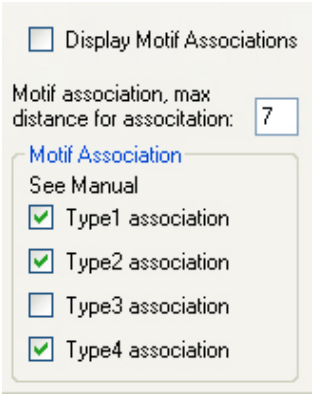
5.2.3 Motif association statistics

Motif association statistics are the most complicated and sophisticated part of SciRoKo and a represent a unique feature.
The term “Motif association“ has been chosen instead of compound microsatellite, because there is no clear definition for the term compound microsatellite and SciRoKo allows flexible adjustment of many parameters affecting compound microsatellite frequency such as the allowed distance between two consecutive microsatellites.
Additionally a compound microsatellite might consist of more than two microsatellites and SciRoKo only works with pairs of neighboring microsatellites (SSR-Couples).
For instance a compound microsatellite consisting of 3 microsatellites will be treated as two motif associations.

For successful compound microsatellite analysis each fasta-identifier (text after the greater than symbol ‘>’) has to be unique.
Since this is always the case with sequences obtained from Genbank we do not expect any complications.
The algorithm used for identification of motif associations first sorts all microsatellites according the fasta identifier and with equal identifier according to the SSR-start position. If the distance between two neighboring SSRs is less or equal to a specified distance the two SSRs are denoted as motif association.

Settings for motif associations: The specified lower and upper boundary also affect the motif association statisitcs, for instance only motif associations between trinucleotide microsatellites might be displayed.
For calculating the total number of Motif associations or the average length of an microsatellite participating in a motif associationit is necessary to set the minimum count value to 1, otherwise the motif associations with a frequency lower than the specified value will not be considered.

Motif association specific settings:



The maximum allowed distance between two neighbouring SSRs for a successful annotation as motif association can be set to a value of choice. The default value is 10. The minimum distance between two adjacent SSR is 1 and not 0.

Displaying motif association statistics

F.Motif	Sec.Motif	Counts	Av.Distance		Av.length.f.	Av.length.sec	Av.MM.Fir. Av.MM.Sec	Counts/Mbp
AAC	AGC	12	1,33	31,83	21,83	2,17	0,58	0,99
AAT	AAC	9	1,67	42,00	26,44	2,78	1,11	0,75
AT	AC	5	1,00	22,00	23,20	0,20	0,60	0,41

Fig. 8: Example of motif association statistic output generated with SciRoKo. Most frequent motif associations of Saccharomyzes cerevisiaea

Column 1: standardized motif, first microsatellite
Column 2: standardized motif second microsatellite
Column 3: Average counts, for this category of motif association
Column 4: Average distance between the two microsatellites
Column 5: Average length for the first microsatellite motif
Column 6: Average length for the second microsatellite motif
Column 7: Average mismatches for the first microsatellite
Column 8: Average mismatches for the second microsatellite
Column 9: Counts per mega base pair

5.2.4 Standardization intensities for motif association:

Unfortunately the standardization of motif associations is a bit complicated because for two adjacent microsatellites a number of configurations must be considered. Each of the two microsatellites forming a motif association can be standardized in two intensities (partially and fully) additionally the conformation and the 5'-3' arrangement has to be considered. Therefore standardization of motif associations requires four standardization intensities compared to only two for microsatellite motifs.

```
5′-AGAGAGAGAGAGAGAGAGTGTGTGTGTGTGTGTGTGTGTG-3′
3′-TCTCTCTCTCTCTCTCTCACACACACACACACACACACACAC-5′
```

Fig.9: Example of a compound microsatellite. For brevity the upper strand will be abstracted as AG-TG and the lower strand as CA-CT

The compound microsatellite in Figure 9 will act as example to demonstrate the different standardization intensities which might be applied to motif association.
The upper strand of the microsatellite in Figure 9 might be written as: 5'-(AG)9-(TG)11-3'
Four our purpose this is still to long therefore this microsatellite motif association will be written as simply: **AG-TG**
The lower strand compound microsatellite will be written as **CA-CT**.
It can easily be seen that AG-TG and CA-CT actually represent the same compound microsatellite, therefore they should not be displayed as different motif associations. That's when we enter the domain of motif association standardization.

In introducing the motif association statistics we will start with the least standardization intensity moving forward to the most intense standardizations

Type 4 motif association:

Type 4 motif associations represent the least intensity of standardization used in SciRoKo. They represent associations of partially standardized microsatellites, the reverse strand compound microsatellite is not considered the 5'-3' arrangement is considered

Microsatellite 1: 5'-GAGAGAGAGAGAGAGAGACTCTCTCTCTCTCT-3'
Microsatellite 2: 5'-AGAGAGAGAGAGAGATCTCTCTCTCTCTCT-3'
Microsatellite 3: 5'-GAGAGAGAGAGAGAGAGTCTCTCTCTCTCT-3'

Fig. 10: Examples of different microsatellites which will be grouped into one category in the type 4 motif association standardization intensity, forming the motif association AG-TC

For type 4 motif associations the two microsatellites forming a SSR-Couple are just partially standardized. Figure 10 demonstrates which microsatellites will be grouped together in this standardization intensity. The two motif associations introduced in Figure 9 AG-TG and CA-CT are not grouped into one category in the type 4 motif associations.
Note that the motif associations AG-CT and CT-AG are not grouped into one category, since the type 4 motif association still consider the 5'-3' arrangements of the two microsatellites forming the SSR-Couple.

TGC	*TTG*	*5*	*1,80*	*19,60*	*26,60*	*0,20*	*1,40*	*0,41*
AAC	*AGC*	*4*	*1,00*	*32,50*	*27,75*	*2,25*	*1,25*	*0,33*
ATT	*TTG*	*3*	*1,00*	*48,67*	*31,00*	*3,00*	*2,33*	*0,25*
AC	*AT*	*3*	*1,00*	*25,33*	*18,33*	*1,00*	*0,33*	*0,25*
TTC	*ATC*	*2*	*1,00*	*22,50*	*40,50*	*0,00*	*3,00*	*0,17*
TTG	*TGC*	*2*	*1,00*	*19,00*	*16,50*	*0,50*	*0,00*	*0,17*
AGC	*AAC*	*1*	*1,00*	*20,00*	*81,00*	*1,00*	*9,00*	*0,08*

Fig. 11: Example of type 4 motif associations identified in Saccharomyzes cerevisaea

Type 3 motif associations:

Type 3 motif associations apply a more vigorous standardization than type 4 motif associations but still not as intense than type 2 motif associations.
Type 3 motif associations represent associations of partially standardized microsatellites, the reverse strand compound microsatellite is considered the 5'-3' arrangement is considered.
Finally the two motif associations introduced in Figure 9 - AG-TG and CA-CT - will be grouped into one category in the type 2 motif associations.

Note that the motif associations AG-CT and CT-AG are still not grouped into one category, since the type 3 motif association still consider the 5'-3' arrangements of the two microsatellites forming the SSR-Couple. See also Figure 12.

```
Microsatellite 1:
5′-AGAGAGAGAGAGAGAGTGTGTGTGTGTGTGTGTGTG-3′
3′-TCTCTCTCTCTCTCTCTCACACACACACACACACACAC-5′

Not equal to microsatellite 2 in type 3 motif associations:
5′-TGTGTGTGTGTGTGTGAGAGAGAGAGAGAGAGAGAG-3′
3′-ACACACACACACACACACTCTCTCTCTCTCTCTCTCTC-5′
```

Fig. 12: Example of two compound microsatellite which are **not** grouped together in type 3 motif associations

Note: the important difference to type 4 motif associations is that type 3 motif associations group the reverse complements strand compound microsatellite into the same category!

AAC	*AGC*	*9*	*1,44*	*29,22*	*23,22*	*1,78*	*0,67*	*0,75*
ATT	*TTG*	*5*	*1,20*	*46,40*	*29,60*	*3,00*	*1,60*	*0,41*
AT	*TG*	*5*	*1,00*	*22,00*	*23,20*	*0,20*	*0,60*	*0,41*
AAT	*AAC*	*4*	*2,25*	*36,50*	*22,50*	*2,50*	*0,50*	*0,33*
TTG	*TGC*	*3*	*1,00*	*39,67*	*17,67*	*3,33*	*0,33*	*0,25*

Fig. 13: Example of type 3 motif associations identified in Saccharomyzes cerevisaea

Type 2 motif associations:

Type 2 motif associations are the next standardization intensity only type 1 motif associations apply a more vigorous standardization intensity. Type 2 motif associations represent associations of partially standardized microsatellites, the reverse strand compound microsatellite is considered the 5'-3' arrangement is ignored.
Therefore the two microsatellites introduced in Figure 12 - AG-TG and TG-AG - will be grouped into one category since type 2 motif associations ignore the 5'-3' arrangement.
Figure 14 demonstrates that type 2 motif associations still consider the conformation of the compound microsatellites therefore the motif associations AG-TG and AG-AC are not grouped into one category in the type 2 motif association.

```
Microsatellite 1:
5′-AGAGAGAGAGAGAGAGAGTGTGTGTGTGTGTGTGTGTGTG-3′
3′-TCTCTCTCTCTCTCTCTCACACACACACACACACACACAC-5′

Not equal to microsatellite 2 in type 2 motif associations:
5′-AGAGAGAGAGAGAGAGAGAGAGACACACACACACACACAC-3′
3′-TCTCTCTCTCTCTCTCTCTCTCTGTGTGTGTGTGTGTGTG-5′
```

Fig. 14: Example of two compound microsatellite which are **not** grouped together in type 2 motif associations

Note: the important difference to type 3 motif associations is that type 2 motif ignore the 5'-3' arrangement grouping motif associations with the same motifs but different 5'-3' arrangements into the same category!

TTG	*TGC*	*12*	*1,33*	*31,83*	*21,83*	*2,17*	*0,58*	*0,99*
ATT	*TTG*	*9*	*1,67*	*42,00*	*26,44*	*2,78*	*1,11*	*0,75*
AT	*TG*	*5*	*1,00*	*22,00*	*23,20*	*0,20*	*0,60*	*0,41*
ATC	*TCG*	*4*	*1,00*	*27,00*	*25,25*	*1,50*	*0,50*	*0,33*
TTC	*ATC*	*3*	*1,00*	*22,00*	*33,67*	*0,00*	*2,00*	*0,25*
AAT	*AGT*	*3*	*2,00*	*40,33*	*25,00*	*3,00*	*1,33*	*0,25*

Fig. 15: Example of type 2 motif associations identified in Saccharomyzes cerevisae!

Type 1 motif associations:

Type 1 motif associations represent the most intense standardization degree. Type 1 motif association represent associations of fully standardize motifs, the 5'-3' arrangement is ignored. The reverse strand compound microsatellite is considerd. The two microsatellites introduced in Figure 14- AG-TG and AG-AC – are grouped into one category in the type 1 motif associations. Type 1 motif associations just represent the fully standardized microsatellite motifs of a SSR-Couple.

AAC	*AGC*	*12*	*1,33*	*31,83*	*21,83*	*2,17*	*0,58*	*0,99*
AAT	*AAC*	*9*	*1,67*	*42,00*	*26,44*	*2,78*	*1,11*	*0,75*
AT	*AC*	*6*	*2,33*	*24,33*	*23,00*	*0,17*	*0,50*	*0,50*
AAT	*ACT*	*5*	*1,80*	*38,20*	*23,80*	*2,40*	*0,80*	*0,41*
ATC	*ACG*	*4*	*1,00*	*27,00*	*25,25*	*1,50*	*0,50*	*0,33*
AT	*A*	*3*	*7,67*	*25,00*	*16,33*	*1,67*	*0,33*	*0,25*
AAG	*ATC*	*3*	*1,00*	*22,00*	*33,67*	*0,00*	*2,00*	*0,25*

Fig. 16 Example of type 1 motif association identified in Saccharoymzes cerevisae!

The motif association standardization pyramid

With each intensification of the standardization degree additional motif associations are grouped into the same category one category, forming a pyramid with type 1 motif associations at the top and type 4 motif associations at the bottom. Figure 17 demonstrates this principle for the most frequent motif association in *Saccharomyzes cerevisae.* Figure 17 demonstrates which motif association are grouped together with progressing standardization intensities.

Type 4:	**Type 3:**	**Type 2:**	**Type 1:**
TGC-TTG AAC-AGC	AAC-AGC	AAC-AGC	AAC-AGC
TTG-TGC AGC-AAC	TTG-TGC		
AAC-TGC AGC-TTG	TGC-AAC	AAC-TGC	
TGC-AAC TTG-AGC	AAC-TGC		

Fig. 17: Standization pyramid for the most frequent motif association of Saccharomyzes cerevisaea