Chapter 2

Selenoprofiles: A Computational Pipeline for Annotation of Selenoproteins

Didac Santesmasses, Marco Mariotti, and Roderic Guigó

Abstract

Selenoproteins contain selenocysteine (Sec or U), the 21st amino acid, inserted in response to an in-frame UGA codon. UGA normally terminates translation, but in selenoprotein mRNAs it is recoded to specify Sec insertion. For this reason, standard gene prediction programs fail to predict Sec codons, and selenoproteins are usually misannotated in protein databases and genome projects. Selenoprofiles is a computational pipeline able to correctly annotate selenoprotein genes in genomic sequences. This program uses a SECIS-independent approach, based on homology searches, and employs curated built-in profile alignments for all known selenoprotein families. Selenoprofiles constitutes the most accurate method for predicting selenoprotein genes belonging to known families.

Key words Selenoprotein prediction, Gene annotation, Selenocysteine, Recoding, UGA codon

1 Introduction

Identification of selenoprotein genes in nucleotide sequences is challenged by the presence of an in-frame UGA codon, normally a stop codon [1]. Selenoprotein transcripts possess an RNA stem-loop structure called SECIS [2] that promotes Sec insertion in response to a UGA codon. For this reason, selenoproteins are generally missed or wrongly annotated by standard gene finding tools. Selenoprofiles [3] is aimed to correctly identify UGA-Sec codons using a homology-based strategy. The key concept is that Sec positions are known a priori in the input selenoprotein family, and thus ad-hoc scoring schemes are employed specifically for the identification of the homologous sites. Selenoprofiles includes a built-in set of manually curated profiles for known selenoprotein families and other proteins related to synthesis of selenoproteins (Table 1), allowing the prediction of these families by a SECIS-independent approach. Selenoprofiles can be used for gene finding in genomes or other nucleotide sequences from species across the whole tree of life. The package also includes additional programs to

Laurent Chavatte (ed.), *Selenoproteins: Methods and Protocols*, Methods in Molecular Biology, vol. 1661,
DOI 10.1007/978-1-4939-7258-6_2,

Table 1
List of selenoprotein built-in profiles

Family ID	Description
Machinery	
SBP2	SECIS binding protein 2
SecS	Selenocysteine Synthase, eukaryotic
eEFsec	Sec specific elongation factor, eukaryotic
pstk	Phosphoseryl-tRNASec kinase
secp43	tRNA Selenocysteine 1 Associated Protein; Trnau1ap
SPS[a]	Selenophosphate synthetase, eukaryotic
Metazoa	
DI	Iodothyronine deiodinases
GPx	Glutathione peroxidases, eukaryotic
TR	Thioredoxin reductases
MsrA	Methionine sulfoxide reductase A, eukaryotic
SelR	Methionine sulfoxide reductase B
Sel15	Selenoprotein 15; SELENOF
Fep15	Fish selenoprotein 15; SELENOE
SelKi	Insect selenoprotein K; SELENOK
SelK	Non-insect selenoprotein K; SELENOK
FrnE	FrnE/DsbA oxidoreductase
SelW	Selenoproteins W and V; SELENOW and SELENOV
AhpC[b]	Alkyl hydroperoxide reductase C
Sel*	Selenoprotein *; SELENO* (* is any of HIJLMNOPSTU)
Protist	
EhSEP2	Emiliania huxleyi disulfide-isomerase like selenoprotein
Lmsel1	Leishmania major selenoprotein 1
SelQ	Toxoplasma gondii selenoprotein Q
SelTryp	Kinetoplastida SelTryp
MSP	Membrane selenoprotein
Ostsp*	Ostreococcus selenoprotein * (* is any of 123)
Sel*	Plasmodium selenoprotein * (* is any of 1234)
Prokarya	
seld	Selenophosphate synthetase, prokaryotic
gpx_b	Glutathione peroxidases, prokaryotic
msra_b	Methionine sulfoxide reductase A, prokaryotic
di_b	Deiodinase-like prokaryotic protein
fdha	FdhA formate dehydrogenase, alpha subunit
frha	FrhA/MvhA/VhuU hydrogenases
frhd	FrhD/MvhD/VhuD hydrogenases
hdra	Heterodisulfide reductase, subunit A
grx	Prokaryotic glutaredoxin 3
gst	glutathione S-transferase
prdb	D-proline reductase, PrdB subunit; selenoprotein B
ars_s	Arsenite S-adenosylmethyltransferase
arsc	Arsenate reductase
bbd	BFD/(2Fe-2S)-binding domain-containing protein
cytc	Cytochrome c-like selenoprotein
dsre	TusD/DsrE sulfurtransferase
duf1858	DUF1858-containing protein (unknown function domain)
fesor	Fe-S oxidoreductase

(continued)

Table 1
(continued)

Family ID	Description
fmdb	FmdB family regulatory protein
ftrb	FtrB ferredoxin thioredoxin reductase beta
frx	Ferredoxin-thioredoxin reductase like
hesb_like	Similar to HesB iron-sulfur cluster biosynthesis protein
imp	IMP dehydrogenase/GMP reductase
mucd	MucD putative serine proteinase
nadh_ox	FAD/NADH-dependent oxidoreductase
pp_sp1	Plesiocystis pacifica Trx-like selenoprotein
prx	Peroxiredoxin family
prx_like	Prx-like thiol:disulfide oxidoreductase
rhor	Rhodanese like sulfurtransferase
rsam	Radical SAM domain-containing oxidoreductase
soret	Split soret cytochrome c precursor
tdip	Thiol:disulfide interchange selenoprotein
ugc	UGC-containing hypothetical selenoprotein
ugsc	UGSC-containing hypothetical selenoprotein
uos_hp3	Hypotheical selenoprotein OS_HP3
usha	UshA UDP-sugar hydrolase
yeee	YeeE/YedE hypothetical selenoprotein
ahp*	Alkyl hydroperoxide reductase subunit * (* is any of df)
dsb*	Thiol-disulfide isomerase subunit * (* is any of ag)
grd*	Glycine reductase complex component * (* is any of ab)
mer*	Mercuric transport protein mer* (* is any of pt)
rnf*	NADH:ubiquinone oxidoreductase subunit * (* is any of bc)

Note that some protein families are split into two profiles for technical reasons (e.g., SelK and SelKi; MsrA and msra_b). The * character is used to indicate multiple families (see the description)
[a]SPS is also part of the metazoan set of profiles
[b]AhpC is also part of the prokaryotic set of profiles

collect and visualize the results in the context of the phylogenetic tree of the species analyzed. This chapter constitutes a practical guide for using Selenoprofiles for selenoprotein search. For more information, see the original publication [3] and the Selenoprofiles manual available in [4].

2 Selenoprofiles Overview

Selenoprofiles is a pipeline combining a number of "slave" programs for homology-based gene finding, whose predictions are analyzed and processed to produce complete multi-exonic gene predictions (Fig. 1). These programs are Blast (psitblastn) [5], Exonerate [6], and Genewise [7]. The three programs are based on the same principle: the target nucleotide sequence is translated in all possible frames and the protein query is aligned to such translated sequences, searching for high scoring matches. The

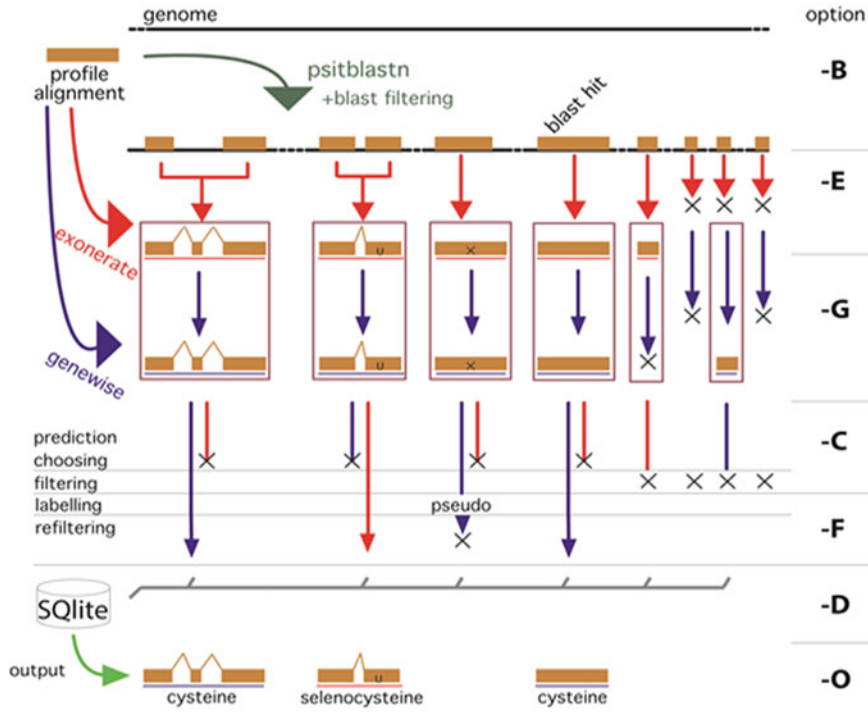

Fig. 1 Graphical summary of the Selenoprofiles pipeline. Reproduced and adapted from [3] by permission of Oxford University Press

program psitblastn is first used to scan the target sequence using a position-specific scoring matrix (PSSM) derived from the input profile alignment. The matches are then used through the two splice alignment programs, Genewise and Exonerate, to deduce the exonic structure of each candidate gene. The best prediction among the three programs is chosen and labeled by a dedicated procedure. Three layers of filtering are used to control the number of potential candidates and to exclude likely false positives. The filtering procedures are highly flexible and customizable, and can be adapted for each input profile independently.

The default filtering procedure is based on sequence similarity. A similarity score threshold is chosen based on the overall conservation present in the input profile sequence alignment; we call it Average Weighted Sequence Identity (AWSI) filter. In practice, this means that protein families with high sequence conservation have strict filtering procedures by default. When multiple profiles are searched, overlapping matches are assigned to one of the profiles, based on sequence similarity. The program produces nonoverlapping multi-exonic gene structures for all input profiles.

3 Installation

Selenoprofiles can be installed on any Unix/Linux system with python 2.6 or higher. All its "slave" programs must be previously installed by the user (*see* **Note 1**): Blast [5], Exonerate [6], Genewise [7], and Mafft [8]. The GNU AWK (GAWK) utility is also needed. The protein database UniRef50 is required for using the built-in profiles (*see* **Note 2**). The following commands should be executed in a terminal to install and test the Selenoprofiles package (*see* **Note 3**). Note that "/install_folder" is used to indicate the hypothetical desired location of installation. Administrator privileges are not required for installing or running Selenoprofiles.

```
cd /install_folder
git clone https://github.com/marco-mariotti/selenoprofiles
cd selenoprofiles
python install_selenoprofiles.py -full
python test_selenoprofiles.py
```

After installation, inside the folder "/install_folder/selenoprofiles," the "Selenoprofiles" executable is available (*see* **Note 4**), as well as a global configuration file named "selenoprofiles_3.config." This file contains all default options and parameters used by the program, each of which can be overridden via a command line option.

4 Input

The program takes two inputs: one or more profile alignments, representing the families to search for; and a "target" genome sequence (or any other nucleotide sequence database) which is to be searched.

The package includes a built-in set of manually curated profiles of known selenoprotein families and proteins related to selenoprotein synthesis. These profiles have been extensively tested and their filtering parameters optimized. The complete list of profiles can be found inside the "profiles/" folder in the Selenoprofiles installation directory. These profiles can be used out-of-the-box with the complete ("-full") installation of Selenoprofiles. Custom user-defined profiles can also be easily created from protein alignments and searched for. For a guide to create new profiles, refer to the Selenoprofiles manual available from [4].

5 Running Selenoprofiles

5.1 Basic Usage

Let us consider that we want to scan the genome of the species *Homo sapiens*, contained in the file "/genomes/Homo_sapiens/genome.fa," for the built-in profile GPx (glutathione peroxidases). Here is the basic command line:

```
Selenoprofiles results -t /genomes/Homo_sapiens/genome.fa -s
"Homo sapiens" -p GPx
```

The first argument (the "results" folder) is where all output and intermediate files are stored. The profile argument "-p" specifies the protein family to search for. Use a comma-separated list to specify more than one family (e.g., "-p GPx,SPS"). Specific option arguments are also accepted to specify convenient sets of profiles: "metazoa," "protist," "prokarya," "machinery" (these are defined in the Selenoprofiles configuration file). For example, this command will search for all selenoprotein profiles previously identified in metazoans genomes and for the Sec machinery (i.e., proteins involved in selenoprotein synthesis):

```
Selenoprofiles results -t /genomes/Homo_sapiens/genome.fa -s
"Homo sapiens" -p metazoa,machinery
```

For RNA sequences or prokaryotic genomes, which are not expected to contain introns, the option "-no_splice" is recommended. This option turns off the procedures for the identification of splicing events, and the pipeline runs faster.

At the end of computation, the Selenoprofiles output files can be found in the output folder located inside the results folder. For the example above, this is "results/Homo_sapiens.genome/output/." The results folder contains also additional intermediate files, which would rarely be inspected by the user. The contents of the output folder are described in the next section. Selenoprofiles attempts to minimize computation by not executing procedures for which the resulting files are detected. When the program is executed again with changed parameters, the user should take care of deleting the previous output files and add an option to specify which pipeline steps must be repeated (*see* Fig. 1). For example, if the final filtering parameters are changed, option "-F" must be provided.

5.2 Output

The output folder contains all output files. For each family with results, a ".ali" file is produced (e.g., "GPx.ali"). This fasta formatted file contains the alignment of all results for this family and the sequences from the profile. The fasta header of the results starts with the "output id" of the prediction, following the structure "family.index.label" (e.g., "GPx.30.selenocysteine"). The "output

Table 2
Output formats available in Selenoprofiles

Format	Description
p2g	Native output format (*see* Fig. 2)
fasta	Protein sequence in fasta
gff	Genomic coordinates in GFF[a]
gtf	Genomic coordinates in GTF[a]
cds	Coding sequence in fasta
dna	The full gene sequence, including introns, in fasta
three_prime	The sequence downstream of the prediction (default 6KB)
five_prime	The sequence upstream of the prediction ("-five_prime_length" must be specified)
introns	The sequence of all introns split in a multi-fasta file

[a]See http://www.ensembl.org/info/website/upload/gff.html for details

id" is a unique identifier for each result in the target, and it is used as a prefix for all output files, using the specific format of the file as a suffix (e.g., "GPx.30.selenocysteine.fasta") (*see* Table 2 for possible output formats). The index included in the "output id" has to be considered an arbitrary identifier; it does not indicate the protein subfamily (e.g., "GPx.4.selenocysteine" is not necessarily the protein known as GPx4). The "output id" also includes a label. The labeling procedure is the following: for selenoprotein families, the label is used to characterize the amino acid aligned to the Sec position. Possible labels are "selenocysteine," "cysteine" or any other amino acid. If the prediction does not span the Sec position in the profiles, the label "unaligned" is used. If it contains frameshifts or in-frame stop codons (apart from the Sec-TGA) the label "pseudo" is used. The label "uga_containing" is used if the only pseudogene feature is one or more TGA codons in any position other than the Sec position. For standard proteins (non-selenoprotein families), the possible labels are "homologue" or "pseudo."

By default, results are provided in the p2g format (Fig. 2), which contains extensive information about the candidate gene. Its header contains basic information about the gene prediction, most of which is self-explanatory. In addition, a few different metrics based on the comparison of the prediction and the sequences in the input profile are provided. The *ASI* corresponds to average sequence identity with the sequences in the profile. *AWSIc* and *AWSIw* are similar, but more sophisticated, scores (see the Selenoprofiles manual available from [4]). Next, the alignment section contains the query-target amino acid sequence alignment, showing the gene structure. Between the query and target

```
Output_id:  GPx.30.cysteine
----------  ---------------
-Target         /genomes/Homo_sapiens/genome.fa
-Chromosome (+) chr6
-Program        genewise
-Query name     Chain A, Crystal Structure Of Human Glutathione Peroxidase 5
-Query range    23-215     length:215   coverage: 0.9
-Profile range  105-445    length:445   coverage: 0.77    sec_position: [192]
-ASI:           0.4008     (ignoring gaps: 0.4602)
-AWSIc:         0.7227     Z-score: 0.726
-AWSIw:         0.7302     Z-score: 0.855
-State          kept

------- alignment -------
Query   MKMDCHKDEKGTIYDYEAIALNKNEYVSFKQYVGKHILFVNVATYUGLTAQYPE <---Intron---> L
         /||||||||||||||||||||||||||||||||||||||||||||||||||| |||||||| <   2173bp   > |
Target  FQMDCHKDEKGTIYDYEAIALNKNEYVSFKQYVGKHILFVNVATYCGLTAQYPE                 L
        tcagtcaggagaatgtggagcaaagtgttactggacactgaggattgcagctcg                  c
        tatagaaaaagctaaaactctaaaaatctaaatgaattttatccaggtccaac    gt     ag   at
        cggtccacgaccctctgccattgtattccggtgcgccccccgcccttgagatt                  aa
                                                  *

Query   NALQEELKPYGLVVLGFPCNQFGKQEPGDNKEILPGLK <---Intron---> YVRPGGGFVPSFQLFEK
        |||||||||||||||||||||||||||||||||||||| <   425bp    > ||||||||||||||||
Target  NALQEELKPYGLVVLGFPCNQFGKQEPGDNKEILPGLK                YVRPGGGFVPSFQLFEK
        agccggcactgcggtgtctactgacgcggaagaccgca                  tgccgggtgcatcctga
        actaaatacagttttgtcgaatgaaacgaaaattcgta   gt     ag    atgcgggttcgtattaa
        tacgggggcttatggctcccatagaaaatcagtttgc                 gtctagaatattcgttga


Query   GDVNGEKEQKVFSFLK <---Intron---> HSCPHPSEILGTFKSISWDPVKVHDIRWNFEKFLVGPDG
        |||||||||||||||| <   1540bp   > ||||||||||||||||||||||||||||||||||||||
Target  GDVNGEKEQKVFSFLK                HSCPHPSEILGTFKSISWDPVKVHDIRWNFEKFLVGPDG
        gggaggagcagtatta                 cttccctgatgatatattgcgagcgactatgatcggcgg
        gatagaaaaattgtta   gt     ag     acgcaccattgctactcgactataatggataatttgcag
        gtgttaaagacctcgg                 cctttctgtgcacatacgctagctcctgctagcgggtta


Query   IPVMRWSHRATVSSVKTDILAYLKQFKT-
        ||||||||||||||||||||||||||||
Target  IPVMRWSHRATVSSVKTDILAYLKQFKTK
        acgacttccgagatgaagacgttactaaa
        tcttggcagcctgctacattcataataca
        ctcgcgccgtgccacgaccggcggacaca

------- positions -------
Exon 1    28497222     28497381
Exon 2    28499555     28499672
Exon 3    28500098     28500197
Exon 4    28501738     28501941

-------- features -------
None
--------- 3' seq --------
Total sequence length available downstream >= 6000
Sequence until first stop codon:
TAG
 *
```

Fig. 2 The p2g format is the default output provided by Selenoprofiles

sequences, bars are used to show identity "|" or similarity "/" of two aligned residues. The selenocysteine position, if present in the alignment, is marked by a "*," and corresponds to a U in the query sequence. If present, predicted in-frame stop codons, and frameshifts are marked in the alignment. Predicted introns are also included. Their length and the donor and acceptor splice sites are indicated. Next, the genomic positions for the exons are reported. The coordinates are 1-based (the first nucleotide in the chromosome is indexed as 1). Finally, the file reports the sequence at the 3′ of the prediction, up to the first stop codon encountered.

Several other output formats are available; they can be specified by using "-output_*format*" in the command line, where *format* is one of the formats listed in Table 2 (e.g., "-output_fasta").

5.3 SECIS Element

Eukaryotic selenoprotein genes possess a SECIS element in the 3′ UTR [2]. The 3′ UTR is not included in the predictions (which include the coding sequence only), but Selenoprofiles provides an option to easily retrieve those sequences from the genome. When specifying "-output_three_prime," Selenoprofiles fetches the sequence found downstream of each prediction, and generates a fasta file with the suffix ".three_prime" in the output folder. The sequence starts with the first nucleotide downstream of the prediction. The length in nucleotides of the retrieved sequence can be specified with the option "-three_prime_length" (6000 by default). The fasta file ".three_prime" can be uploaded to the Seblastian web server (see [9] and chapter 1 in this book) to search for eukaryotic SECIS elements. Since the SECIS must be in the same strand as the gene, the option "search also complementary strand" in the web server should be unchecked.

6 Searching Multiple Target Genomes

Selenoprofiles can be used to analyze multiple genomes and compare their results. The genomes have to be analyzed in different runs, but the results folder should be the same. For example:

```
Selenoprofiles results -t /genomes/Homo_sapiens/genome.fa -s
"Homo sapiens" -p metazoa,machinery
Selenoprofiles results -t /genomes/Drosophila_melanogaster/
genome.fa -s "Drosophila melanogaster" -p metazoa,machinery
```

After running the pipeline for all the targets, the program selenoprofiles_join_alignments.py is used to search for and to join all the ".ali" files inside each target subfolder.

```
selenoprofiles_join_alignments.py -d results -o joined_align-
ments
```

The command line above will join all the alignments inside the results directory ("-d" option), and will generate the new joined alignments in the output folder ("-o" option). Then, the program selenoprofiles_tree_drawer.py can be used to visualize the results for multiple target species, placed in the corresponding phylogenetic tree. This program requires the python library for tree exploration ETE (http://etetoolkit.org/). The program selenoprofiles_tree_drawer.py takes as input the joined alignments, which are provided as a list of multiple arguments separated by a blank space (the order in which the alignments are provided is used to show them in the plot). The tree of the investigated species is also required (*see* **Note 5**). It is specified using the "-t" parameter, which must be placed as the last argument in the command line. For example:

```
cd joined_alignments
selenoprofiles_tree_drawer.py eEFsec.ali SelU.ali -t species_tree.nw
```

The output of selenoprofiles_tree_drawer.py is the graphical representation of the phylogenetic tree annotated with the results obtained by Selenoprofiles (Fig. 3). Running "selenoprofiles_tree_drawer.py -h" will show additional options for the program.

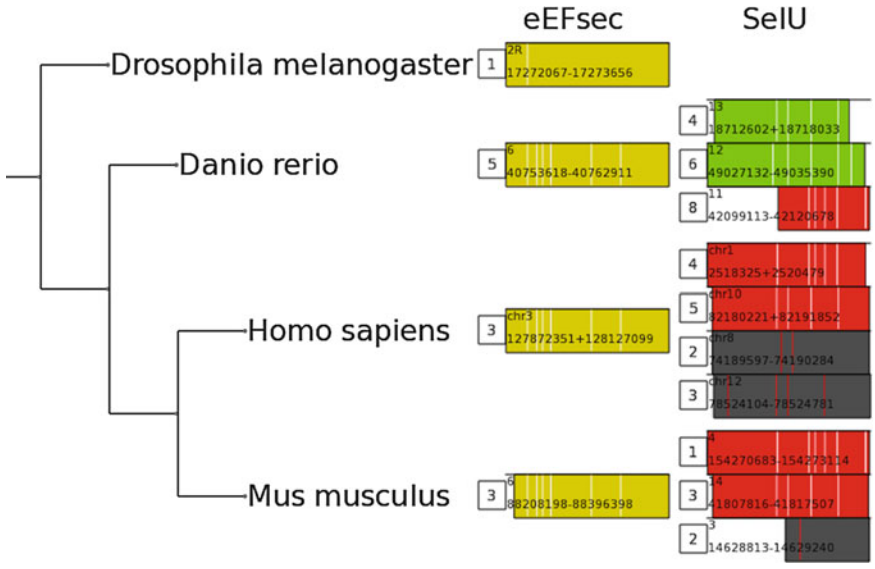

Fig. 3 Species tree annotated with Selenoprofiles results. Each column corresponds to a family, and each colored rectangle corresponds to a result. Multiple results for the same species and family are presented as piled up rectangles. The numeric tag on the left corresponds to the Selenoprofiles numeric index. The color depends on the label: *green* for selenocysteine; *red* for cysteine; *yellow* for homologue (non-selenoprotein families); *gray* for pseudo. Other colors are used for additional labels. The rectangle length and position indicate the coverage of the result aligned to the profile. The information inside the rectangle corresponds to the chromosome, genomic coordinates, and strand. The position of the introns, relative to the protein alignment, is indicated with *vertical white lines*. *Vertical red lines* correspond to insertions causing frameshifts

For further analyses using the alignments produced by selenoprofiles_join_alignments.py, it is strongly recommended to realign the sequences with a multiple sequence alignment program, such as T-Coffee [10] (http://tcoffee.crg.cat/).

7 Notes

1. The following web page contains information on how to install the "slave" programs and possible problems you may encounter during the installation: http://big.crg.cat/news/20110616/installing_programs_and_modules_needed_by_selenoprofiles.
2. The script install_selenoprofiles.py, if executed in "-full" mode, will attempt to download the protein database UniRef50 through the Internet. If the database is already in your system, you can skip this step by specifying its location with the option "-db." Running "install_selenoprofiles.py -h" will show more information.
3. If the program GIT is not installed in your system, you can point your web browser to https://github.com/marco-mariotti/selenoprofiles and download the package manually.
4. In order to run Selenoprofiles and the other programs provided with the package, the "selenoprofiles" folder needs to be present in your $PATH variable. You have to type "export PATH=$PATH:/install_folder/selenoprofiles" (where *install_folder* is the folder where you installed the package) every time you open a new terminal, or include that line in the ".bashrc" file in your home directory.
5. The most commonly used format for phylogenetic trees is the newick format, where the nodes of the tree are defined by parentheses. The tree used in Fig. 3 would be "(Drosophila melanogaster (Danio rerio (Mus musculus, Homo sapiens)));". ETE toolkit (http://etetoolkit.org) provides with automated tools to extract species trees from the NCBI taxonomy database and write them into a newick format file.

References

1. Driscoll DM, Chavatte L (2004) Finding needles in a haystack. In silico identification of eukaryotic selenoprotein genes. EMBO Rep 5:140–141. doi:10.1038/sj.embor.7400080
2. Berry MJ, Banu L, Chen YY et al (1991) Recognition of UGA as a selenocysteine codon in type I deiodinase requires sequences in the 3′ untranslated region. Nature 353:273–276. doi:10.1038/353273a0
3. Mariotti M, Guigó R (2010) Selenoprofiles: profile-based scanning of eukaryotic genome sequences for selenoprotein genes. Bioinformatics 26:2656–2663. doi:10.1093/bioinformatics/btq516
4. Mariotti M (2016) Selenoprofiles 3 | Bioinformatics and Genomics @ CRG http://big.crg.cat/services/selenoprofiles. Accessed 1 Nov 2016

5. Altschul SF, Madden TL, Schäffer AA et al (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25:3389–3402
6. Slater GSC, Birney E (2005) Automated generation of heuristics for biological sequence comparison. BMC Bioinformatics 6:31. doi:10.1186/1471-2105-6-31
7. Birney E, Clamp M, Durbin R (2004) GeneWise and Genomewise. Genome Res 14:988–995. doi:10.1101/gr.1865504
8. Katoh K, Misawa K, Kuma K, Miyata T (2002) MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. Nucleic Acids Res 30:3059–3066. doi:10.1093/nar/gkf436
9. Mariotti M, Lobanov AV, Guigo R, Gladyshev VN (2013) SECISearch3 and Seblastian: new tools for prediction of SECIS elements and selenoproteins. Nucleic Acids Res 41:e149. doi:10.1093/nar/gkt550
10. Notredame C, Higgins DG, Heringa J (2000) T-coffee: a novel method for fast and accurate multiple sequence alignment. J Mol Biol 302:205–217. doi:10.1006/jmbi.2000.4042

http://www.springer.com/978-1-4939-7257-9

Selenoproteins
Methods and Protocols
Chavatte, L. (Ed.)
2018, XV, 340 p. 63 illus., 35 illus. in color., Hardcover
ISBN: 978-1-4939-7257-9
A product of Humana Press