PYCOEVOL:
A Python workflow to study protein-protein coevolution

Fábio Madeira and Ludwig Krippahl

CENTRIA-DI, **Universidade Nova de Lisboa**, Caparica, Portugal

Universidade Nova de Lisboa

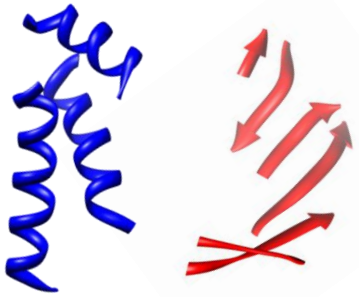
Outline

1. Background
2. Motivation
3. Implementation
4. Results
5. Conclusions
6. Future work

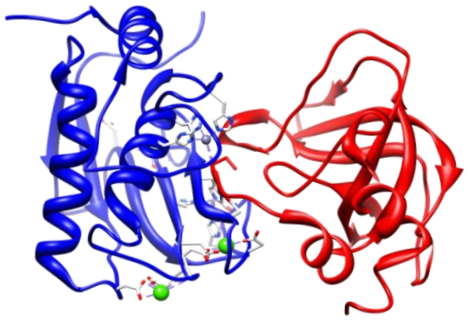
From sequence to structure

DPSLDRPFISEGTTLKDLIYDMTT

Primary structure



Secondary structure



Tertiary/quaternary structure

Protein coevolution

- Co-adaptation or correlated mutations

Protein coevolution

- Co-adaptation or correlated mutations
- Change of a biological object triggered by the change of a related object

Protein coevolution

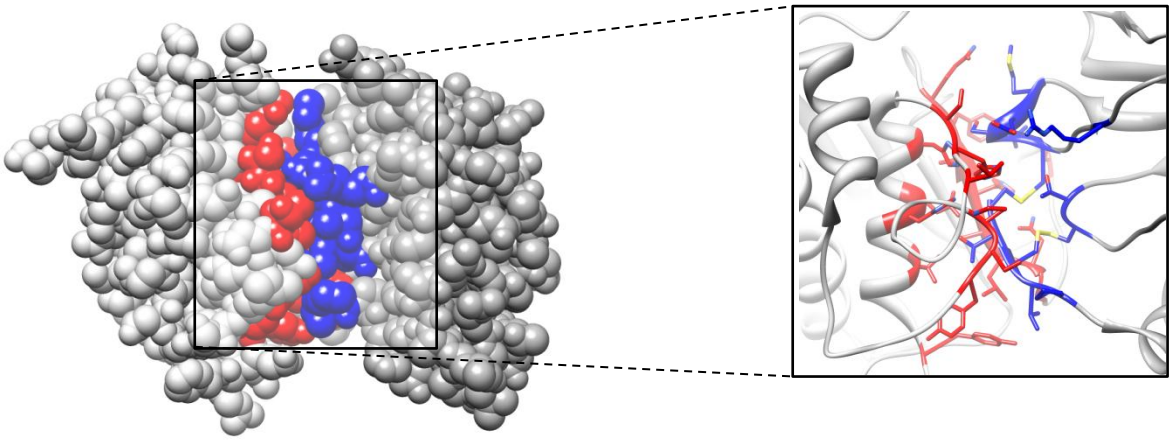
- Co-adaptation or correlated mutations
- Change of a biological object triggered by the change of a related object
- Accumulation of structural and functional changes throughout lineages, imposed by evolutionary constraints

Protein coevolution

- Co-adaptation or correlated mutations
- Change of a biological object triggered by the change of a related object
- Accumulation of structural and functional changes throughout lineages, imposed by evolutionary constraints
- Proposed to aid:
  - Protein folding and fold recognition
  - **Prediction of protein-protein interactions *[Pazos et al, 1997]***

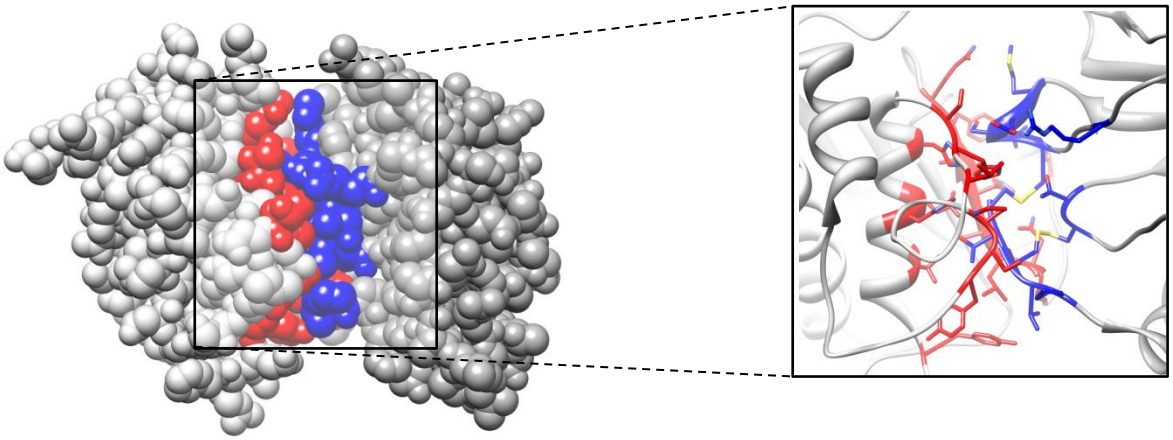
…

Prediction of protein-protein interactions



- Interacting proteins have a key role in every biological process, and they have evolved performing mutual interactions

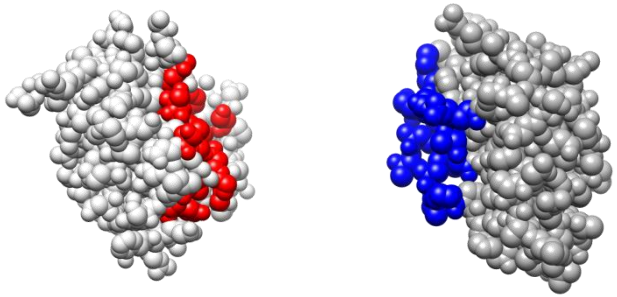
Prediction of protein-protein interactions



- Interacting proteins have a key role in every biological process, and they have evolved performing mutual interactions
- The sequence of an interacting protein must reflect the process of adaptation

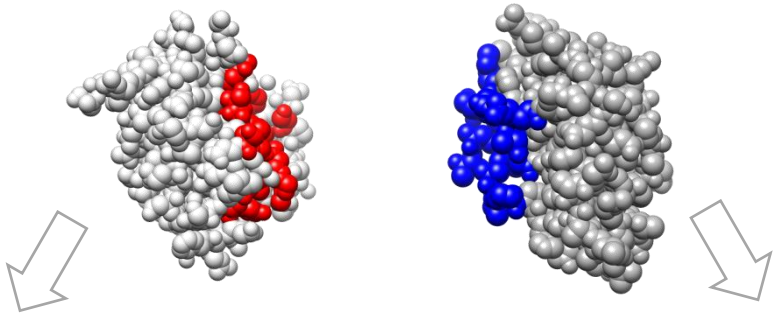
The "classic" coevolution workflow

1. **Search for homologous sequences**



The "classic" coevolution workflow

1. **Search for homologous sequences**

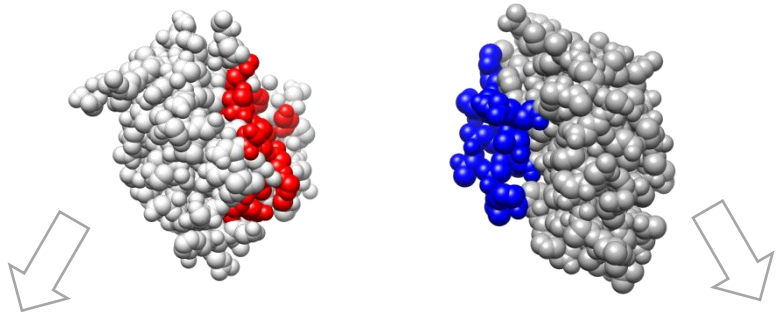


`DPSLDRPFISEGTTLKDLIYDMTT`

`VEGMIKLALSTASGLAHLHMEI`

The "classic" coevolution workflow

1. Search for homologous sequences



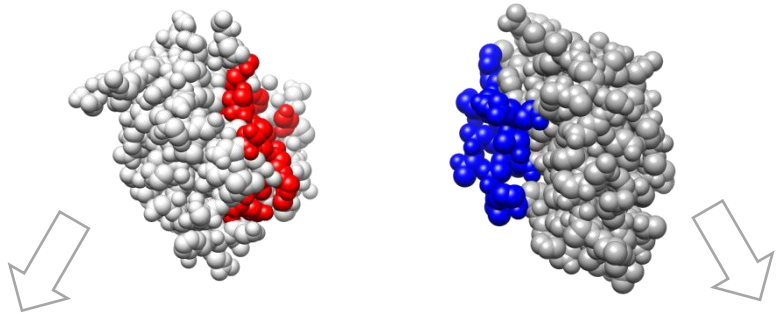
`DPSLDRPFISEGTTLKDLIYDMTT`

`VEGMIKLALSTASGLAHLHMEI`

2. MSA computation

The "classic" coevolution workflow

1. Search for homologous sequences



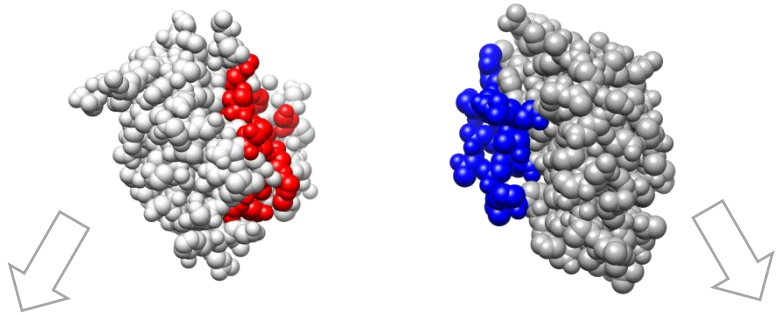
2. MSA computation

```
DPSLDRPFISEGTTLKDLIYDMTT
-----EPRPDSGRDWSVELQEL--
-----KPGPDLGRDWSVELQEL--
-----EPESDSGRDWSAELPEL--
-----EPEPGSGGDCSEELPEL--
-----DPEPGSGGDCSEELPEL--
PEPEQEPEPDSGGDCSAELPEL--
RKQGLHSMNMMEAACSEPSLDL--
```

```
VEGMIKLALSTASGLAHLHMEI
WGSSLRMALSLAQGLAFLHEER
WGSSLRMALSLAQGLAFLHEER
WGSSLRMALSLAQGLAFLHEER
WGSSLSMALSLAEGLAFLHGRR
WGSSLSMALSLAEGLAFLHERR
--SSMSMALSLAQGLAFLHER-
--SSCRLAHSITRGLAYLHTRR
```

The "classic" coevolution workflow

1. Search for homologous sequences



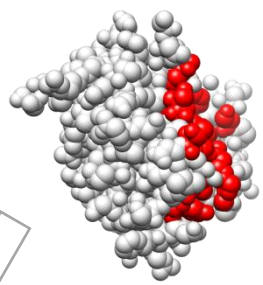
```
DPSLDRPFISEGTTLKDLIYDMTT
-----EPRPDSGRDWSVELQEL--
-----KPGPDLGRDWSVELQEL--
-----EPESDSGRDWSAELPEL--
-----EPEPGSGGDCSEELPEL--
-----DPEPGSGGDCSEELPEL--
PEPEQEPEPDSGGDCSAELPEL--
RKQGLHSMNMMEAACSEPSLDL--
```

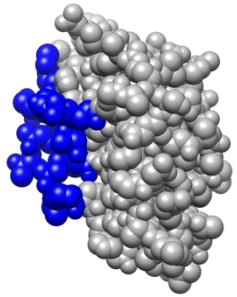
2. MSA computation

```
VEGMIKLALSTASGLAHLHMEI
WGSSLRMALSLAQGLAFLHEER
WGSSLRMALSLAQGLAFLHEER
WGSSLRMALSLAQGLAFLHEER
WGSSLSMALSLAEGLAFLHGRR
WGSSLSMALSLAEGLAFLHERR
--SSMSMALSLAQGLAFLHER-
--SSCRLAHSITRGLAYLHTRR
```

3. Coevolution analysis

The "classic" coevolution workflow



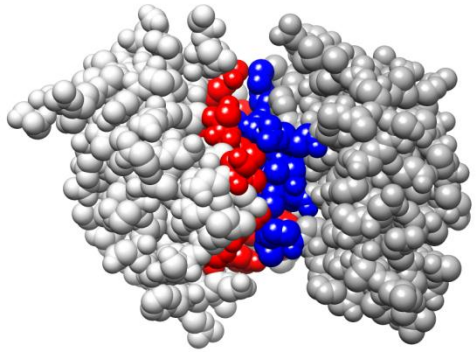


1. Search for homologous sequences

```
DPSLDRPFISEGTTLKDLIYDMTT
-----EPRPDSGRDWSVELQEL--  i
-----KPGPDLGRDWSVELQEL--  j
-----EPESDSGRDWSAELPEL--
-----EPEPGSGGDCSEELPEL--
-----DPEPGSGGDCSEELPEL--
PEPEQEPEPDSGGDCSAELPEL--
RKQGLHSMNMMEAACSEPSLDL--
          x
```

2. MSA computation

```
VEGMIKLALSTASGLAHLHMEI
WGSSLRMALSLAQGLAFLHEER  i
WGSSLRMALSLAQGLAFLHEER  j
WGSSLRMALSLAQGLAFLHEER
WGSSLSMALSLAEGLAFLHGRR
WGSSLSMALSLAEGLAFLHERR
--SSMSMALSLAQGLAFLHER-
--SSCRLAHSITRGLAYLHTRR
     y
```



3. Coevolution analysis

Methods for finding protein coevolution

- *Matrix-based*
  - CLM, CPVN, VOL, etc.
- *Correlation-based*
  - Pearson, Spearman, McBASC, Quartets, OMES, etc.
- *Perturbation-based*
  - MI, SCA, ELSC, etc.
- *Phylogenetic dependent*
  - Mirrortree, etc.

Pycoevol

- Large number of scoring functions

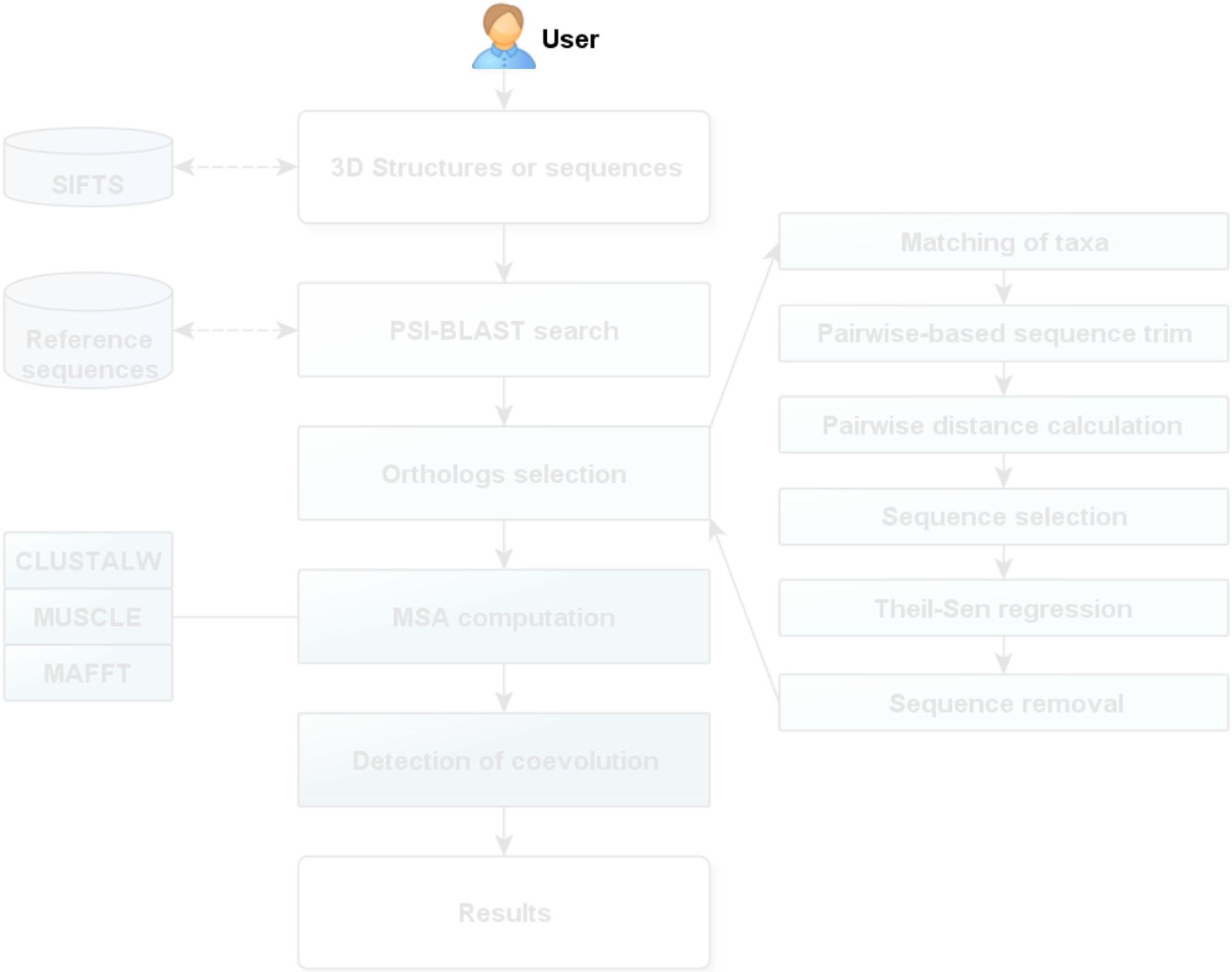
Pycoevol

- Large number of scoring functions
- There is no benchmark or survey on the accuracy of each method

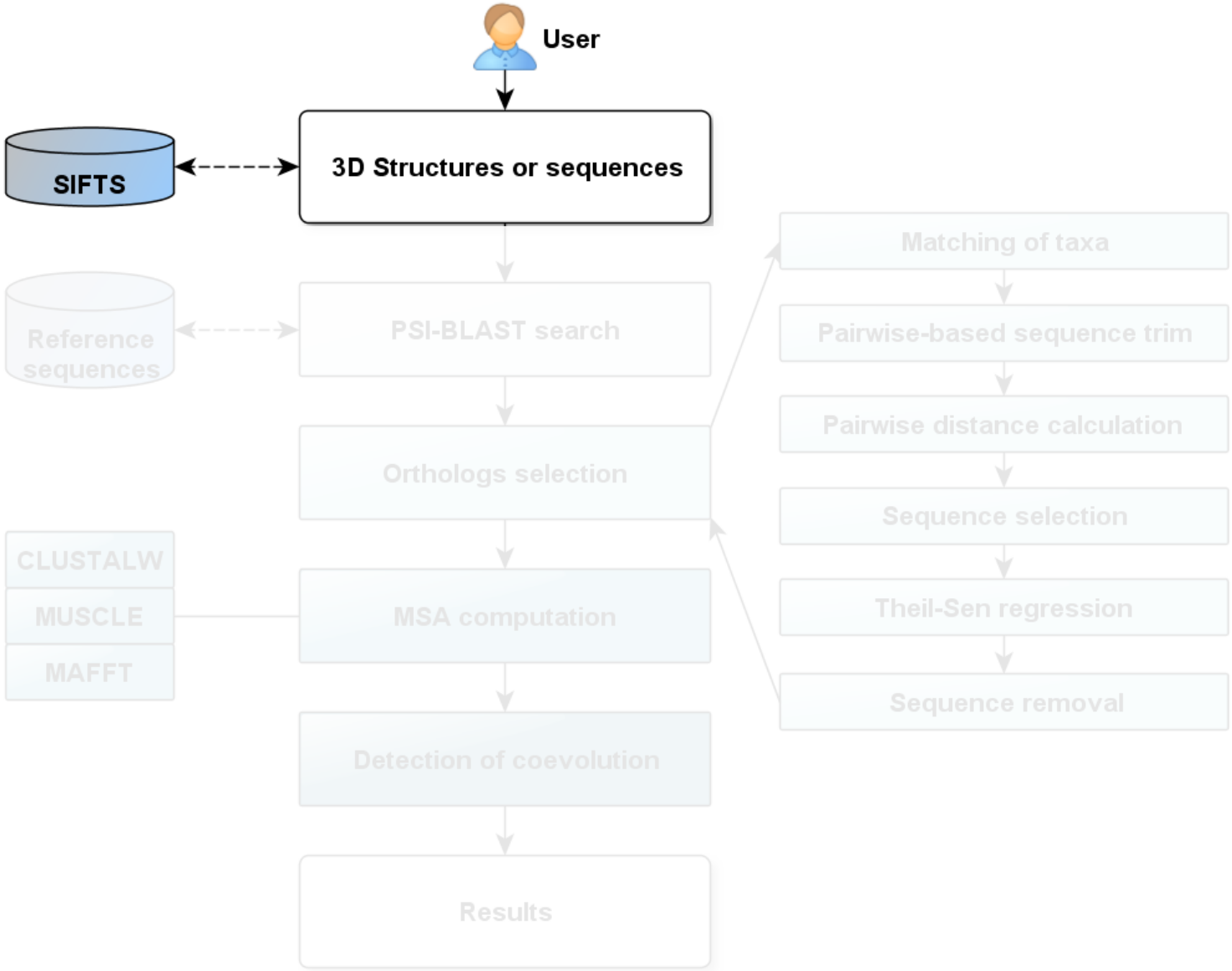
Pycoevol

- Large number of scoring functions
- There is no benchmark or survey on the accuracy of each method
- Most of these methods are not available to the community

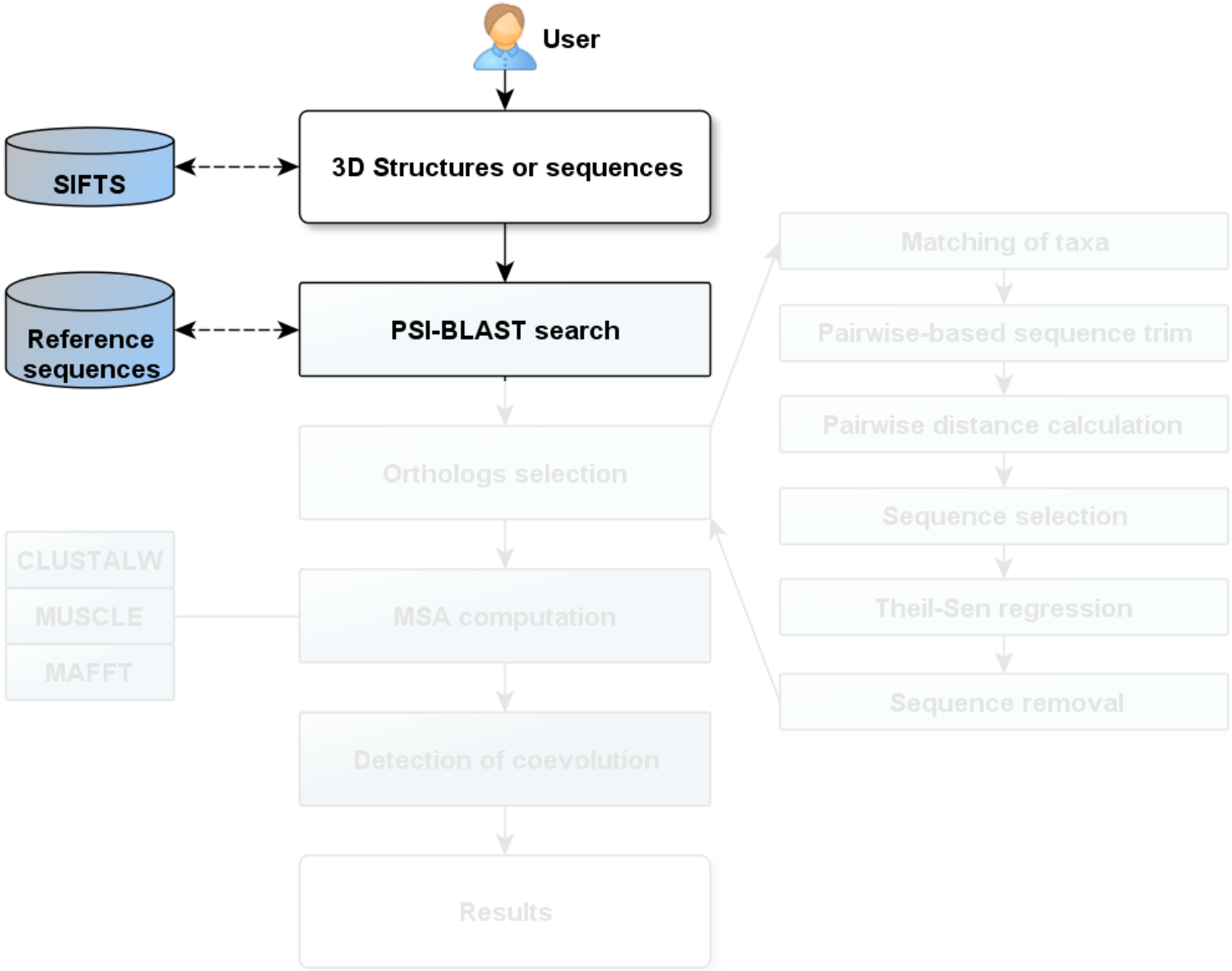
The workflow of Pycoevol



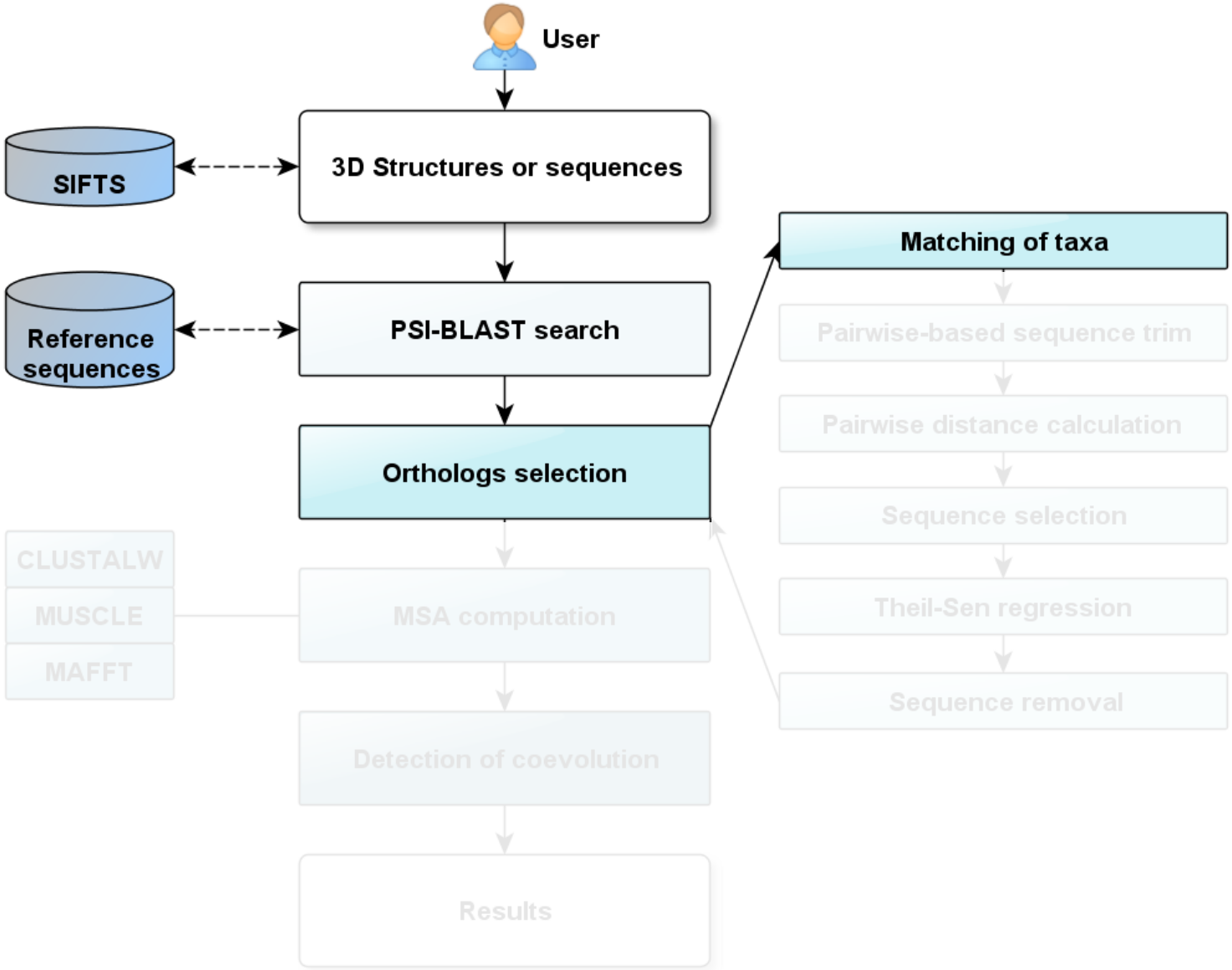
The workflow of Pycoevol



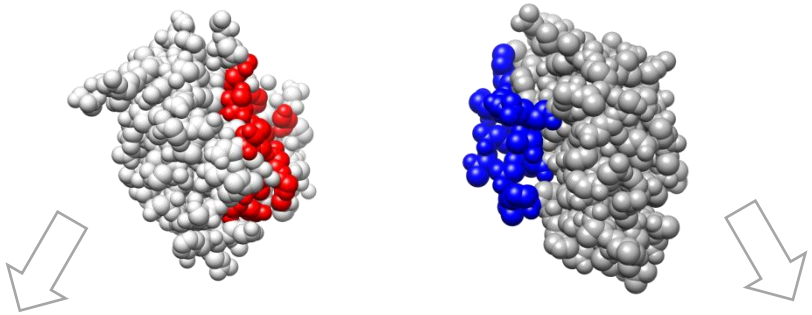
The workflow of Pycoevol



The workflow of Pycoevol



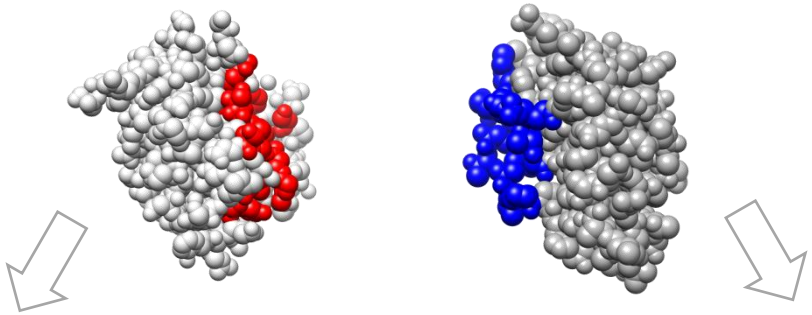
Matching of taxa



```
Organism1  DPSLDRPFISEGTTLKDLIYDMTT
Organism2  -----EPRPDSGRDWSVELQEL--
Organism3  -----KPGPDLGRDWSVELQEL--
Organism4  -----EPESDSGRDWSAELPEL--
Organism5  -----EPEPGSGGDCSEELPEL--
Organism6  -----DPEPGSGGDCSEELPEL--
Organism7  PEPEQEPEPDSGGDCSAELPEL--
Organism8  RKQGLHSMNMMEAACSEPSLDL--
```

```
Organism3  VEGMIKLALSTASGLAHLHMEI
Organism1  WGSSLRMALSLAQGLAFLHEER
Organism9  WGSSLRMALSLAQGLAFLHEER
Organism8  WGSSLRMALSLAQGLAFLHEER
Organism5  WGSSLSMALSLAEGLAFLHGRR
Organism10 WGSSLSMALSLAEGLAFLHERR
Organism2  --SSMSMALSLAQGLAFLHER-
Organism11 --SSCRLAHSITRGLAYLHTRR
```

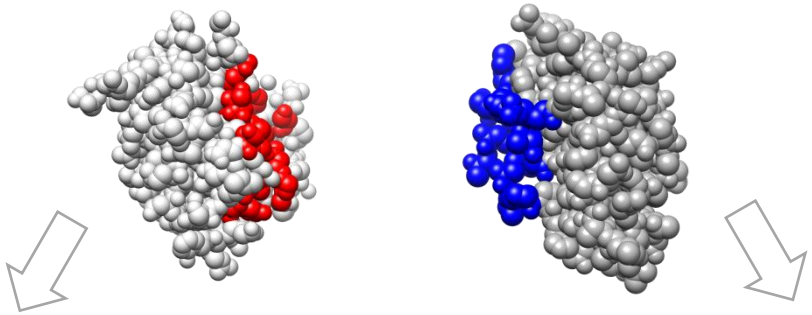
Matching of taxa



Organism1	DPSLDRPFISEGTTLKDLIYDMTT
Organism2	-----EPRPDSGRDWSVELQEL--
Organism3	-----KPGPDLGRDWSVELQEL--
Organism4	-----EPESDSGRDWSAELPEL--
Organism5	-----EPEPGSGGDCSEELPEL--
Organism6	-----DPEPGSGGDCSEELPEL--
Organism7	PEPEQEPEPDSGGDCSAELPEL--
Organism8	RKQGLHSMNMMEAACSEPSLDL--

Organism3	VEGMIKLALSTASGLAHLHMEI
Organism1	WGSSLRMALSLAQGLAFLHEER
Organism9	WGSSLRMALSLAQGLAFLHEER
Organism8	WGSSLRMALSLAQGLAFLHEER
Organism5	WGSSLSMALSLAEGLAFLHGRR
Organism10	WGSSLSMALSLAEGLAFLHERR
Organism2	--SSMSMALSLAQGLAFLHER-
Organism11	--SSCRLAHSITRGLAYLHTRR

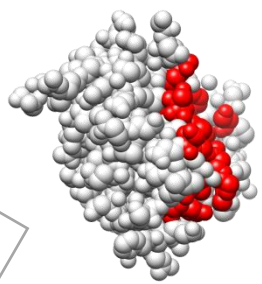
Matching of taxa

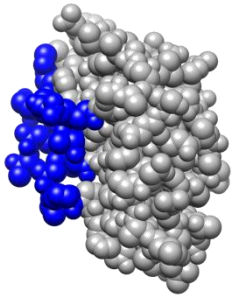


Organism1	DPSLDRPFISEGTTLKDLIYDMTT
Organism2	-----EPRPDSGRDWSVELQEL--
Organism3	-----KPGPDLGRDWSVELQEL--
~~Organism4~~	~~-----EPESDSGRDWSAELPEL--~~
Organism5	-----EPEPGSGGDCSEELPEL--
~~Organism6~~	~~-----DPEPGSGGDCSEELPEL--~~
~~Organism7~~	~~PEPEQEPEPDSGGDCSAELPEL--~~
Organism8	RKQGLHSMNMMEAACSEPSLDL--

Organism3	VEGMIKLALSTASGLAHLHMEI
Organism1	WGSSLRMALSLAQGLAFLHEER
~~Organism9~~	~~WGSSLRMALSLAQGLAFLHEER~~
Organism8	WGSSLRMALSLAQGLAFLHEER
Organism5	WGSSLSMALSLAEGLAFLHGRR
~~Organism10~~	~~WGSSLSMALSLAEGLAFLHERR~~
Organism2	--SSMSMALSLAQGLAFLHER-
~~Organism11~~	~~--SSCRLAHSITRGLAYLHTRR~~

Matching of taxa





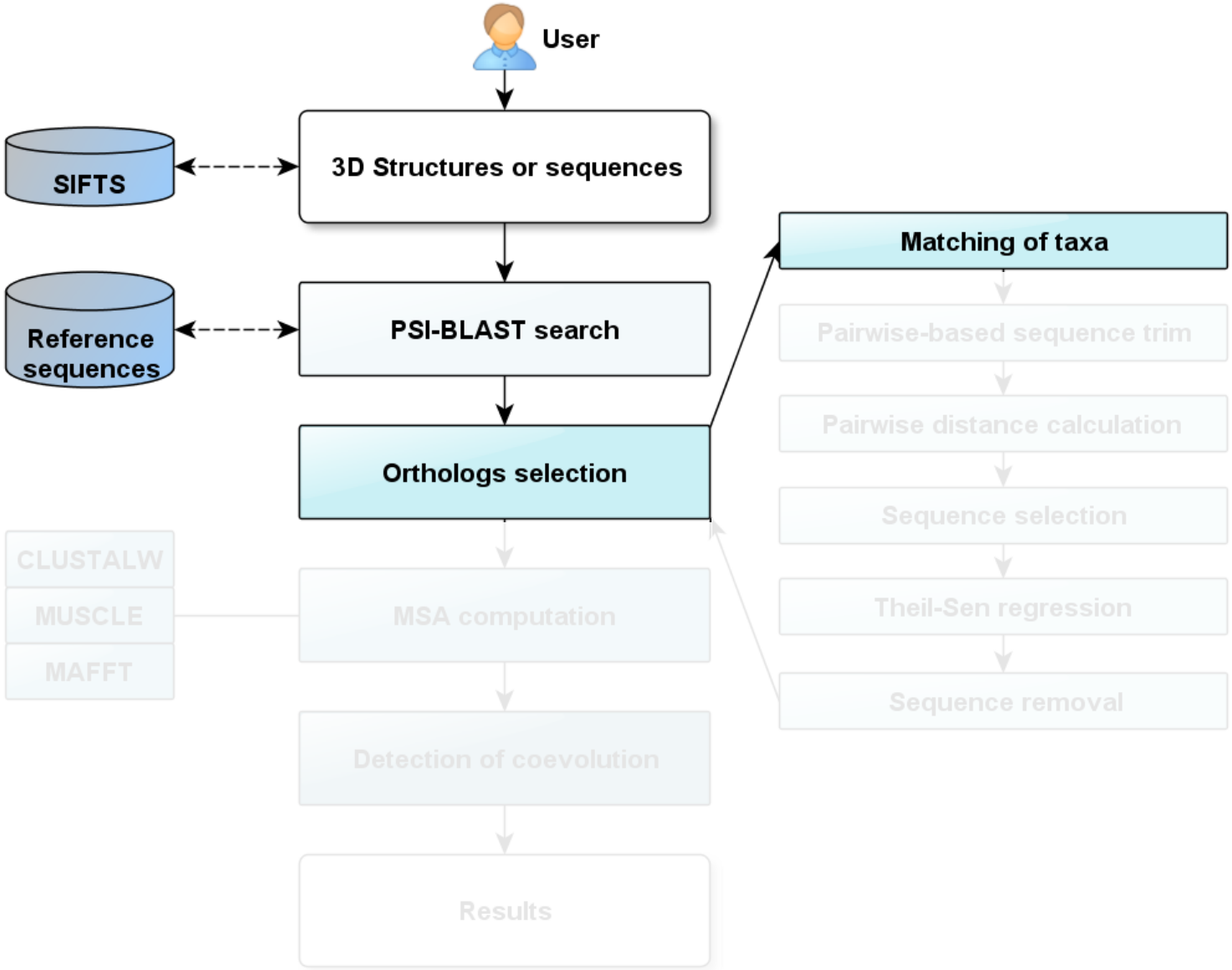
```
Organism1  DPSLDRPFISEGTTLKDLIYDMTT
Organism2  -----EPRPDSGRDWSVELQEL--
Organism3  -----KPGPDLGRDWSVELQEL--
~~Organism4~~  ~~-----EPESDSGRDWSAELPEL--~~
Organism5  -----EPEPGSGGDCSEELPEL--
~~Organism6~~  ~~-----DPEPGSGGDCSEELPEL--~~
~~Organism7~~  ~~PEPEQEPEPDSGGDCSAELPEL--~~
Organism8  RKQGLHSMNMMEAACSEPSLDL--
```

```
Organism3  VEGMIKLALSTASGLAHLHMEI
Organism1  WGSSLRMALSLAQGLAFLHEER
~~Organism9~~  ~~WGSSLRMALSLAQGLAFLHEER~~
Organism8  WGSSLRMALSLAQGLAFLHEER
Organism5  WGSSLSMALSLAEGLAFLHGRR
~~Organism10~~  ~~WGSSLSMALSLAEGLAFLHERR~~
Organism2  --SSMSMALSLAQGLAFLHER-
~~Organism11~~  ~~--SSCRLAHSITRGLAYLHTRR~~
```

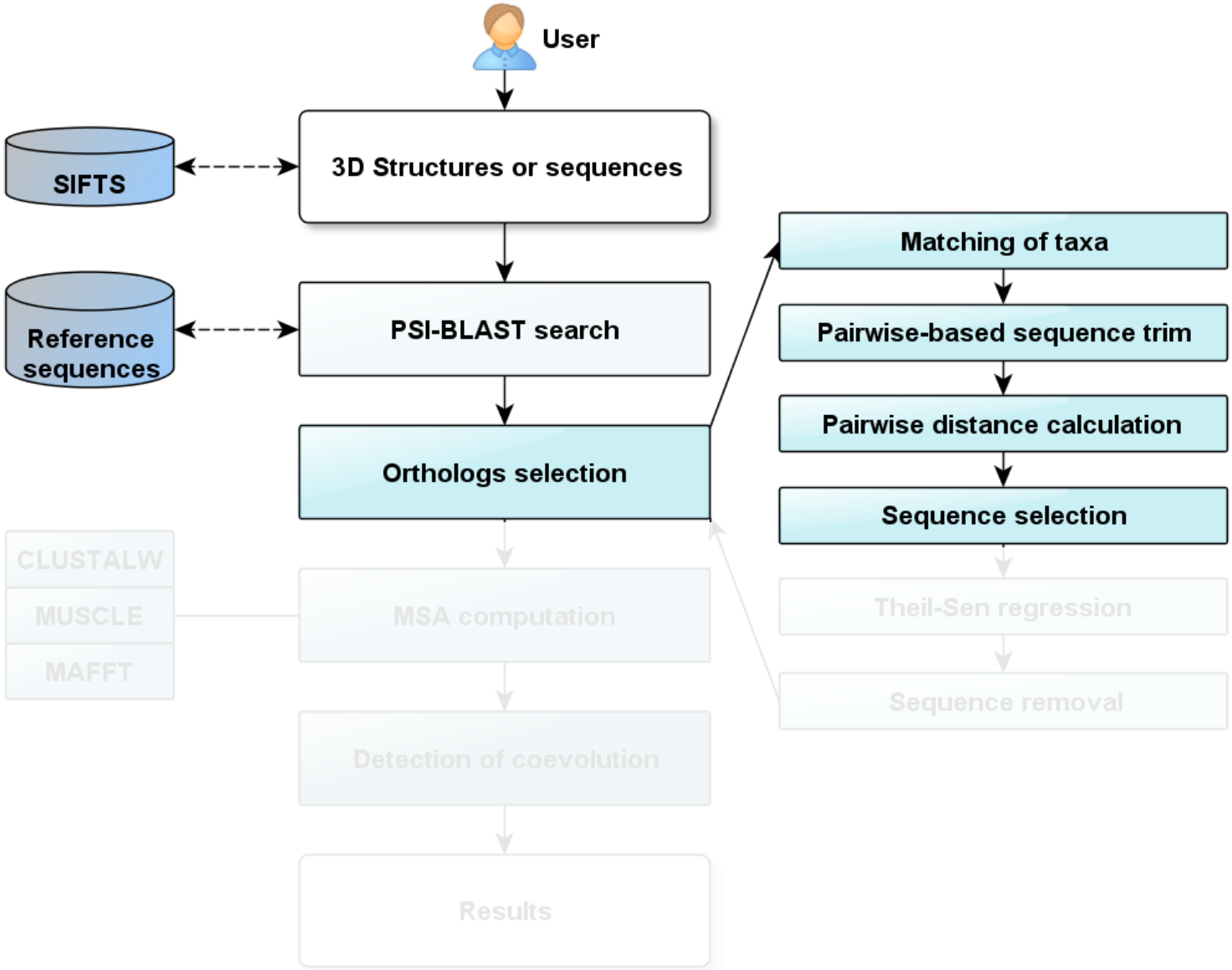
```
Organism1  DPSLDRPFISEGTTLKDLIYDMTT
Organism3  -----KPGPDLGRDWSVELQEL--
Organism2  -----EPRPDSGRDWSVELQEL--
Organism5  -----EPEPGSGGDCSEELPEL--
Organism8  RKQGLHSMNMMEAACSEPSLDL--
```

```
Organism1  WGSSLRMALSLAQGLAFLHEER
Organism3  VEGMIKLALSTASGLAHLHMEI
Organism2  --SSMSMALSLAQGLAFLHER-
Organism5  WGSSLSMALSLAEGLAFLHGRR
Organism8  WGSSLRMALSLAQGLAFLHEER
```

The workflow of Pycoevol



The workflow of Pycoevol



Pairwise distances and orthologs selection

- p-distance *[Jukes and Cantor, 1969]*

$$d = p = \frac{N_{dif}}{N_{total}}$$

- Jukes-Cantor *[Jukes and Cantor, 1969]*

$$d = -\frac{19}{20} log(1 - p * \frac{20}{19})$$

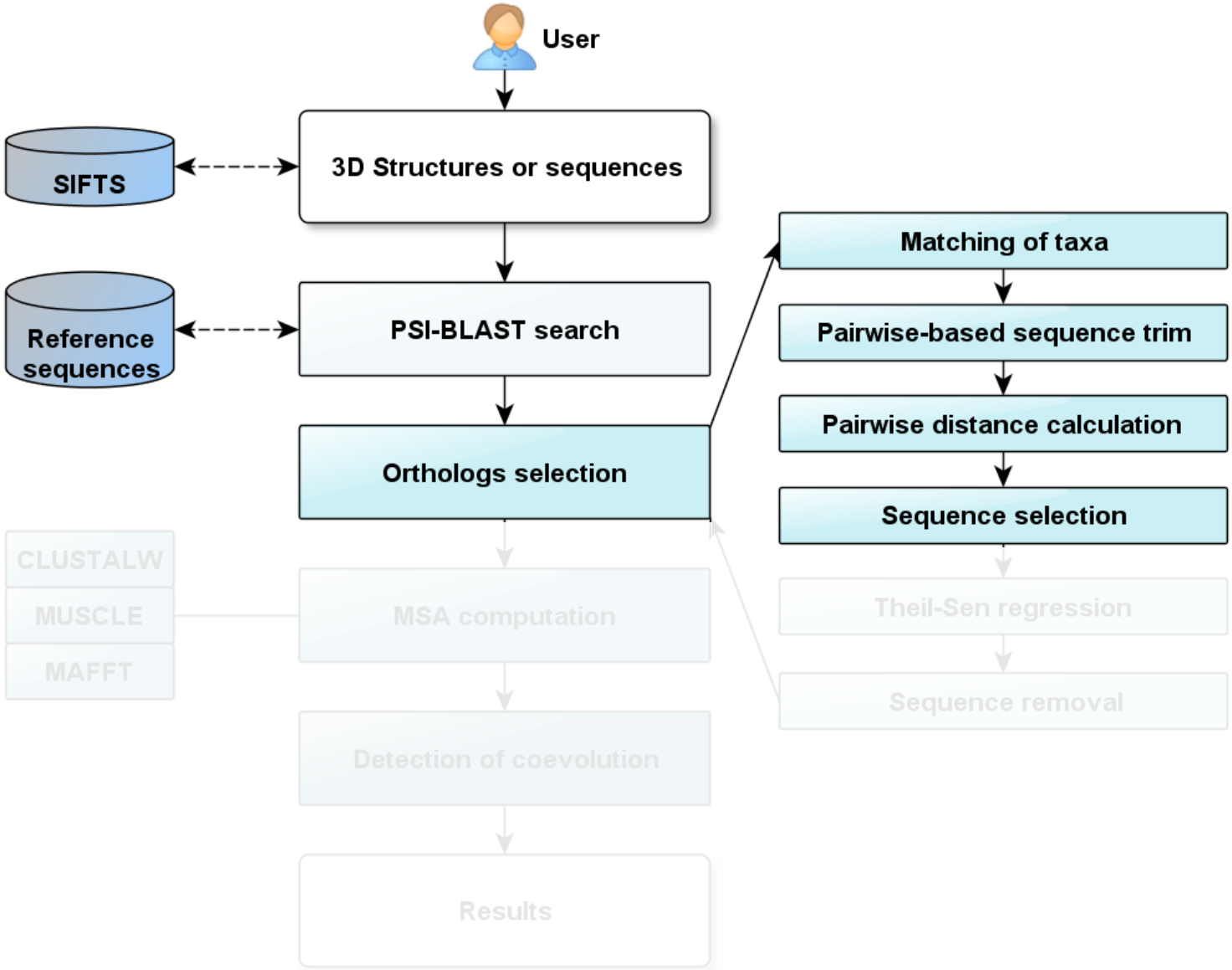
- Kimura distance *[Kimura, 1983]*

$$d = -ln(1 - p - 0.2^2)$$

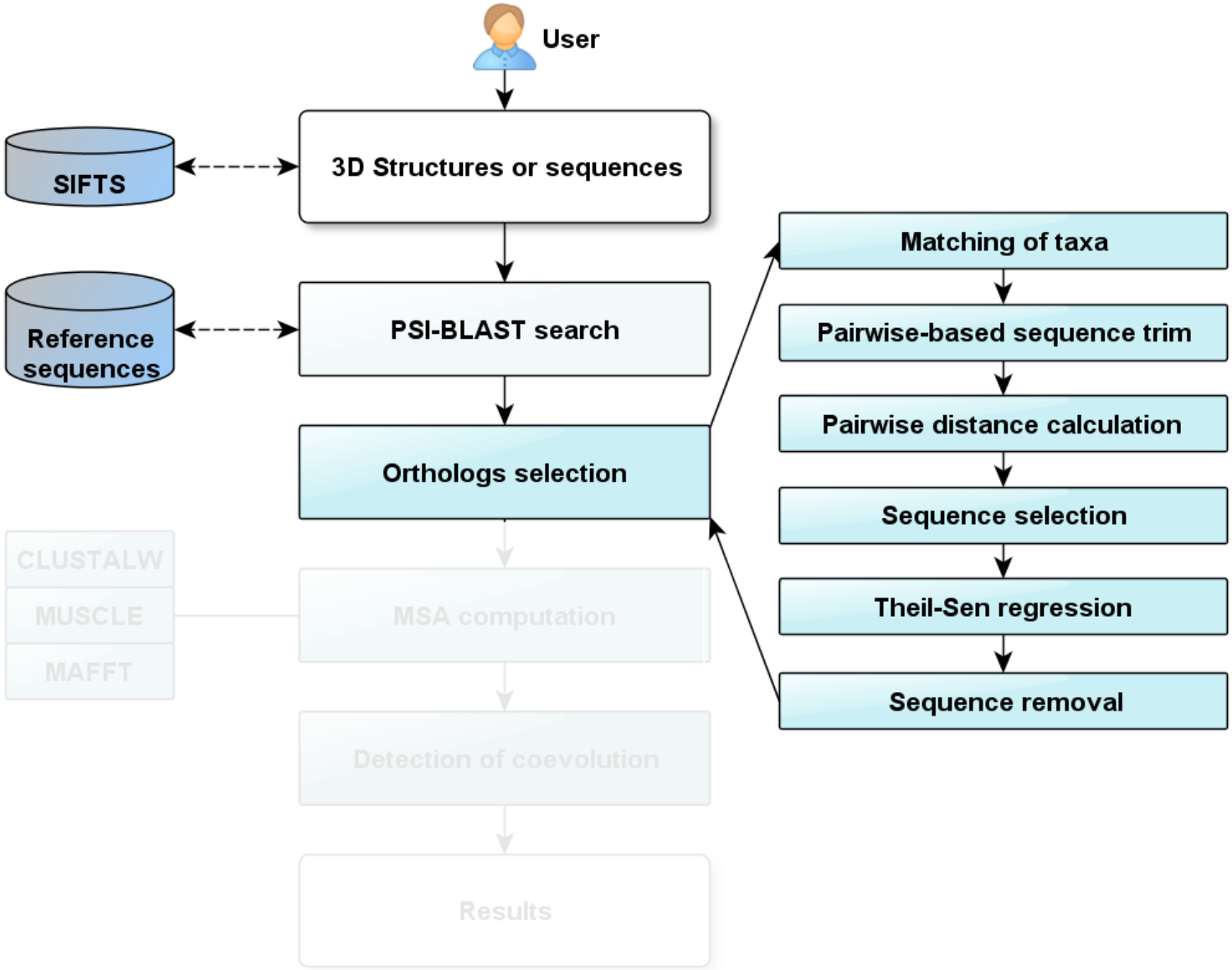
- Pairwise score using Dayhoff matrices

$$d = \sum \frac{1 - S_{ij}}{S_{ii}} * \frac{1 - S_{ij}}{S_{jj}}$$

The workflow of Pycoevol

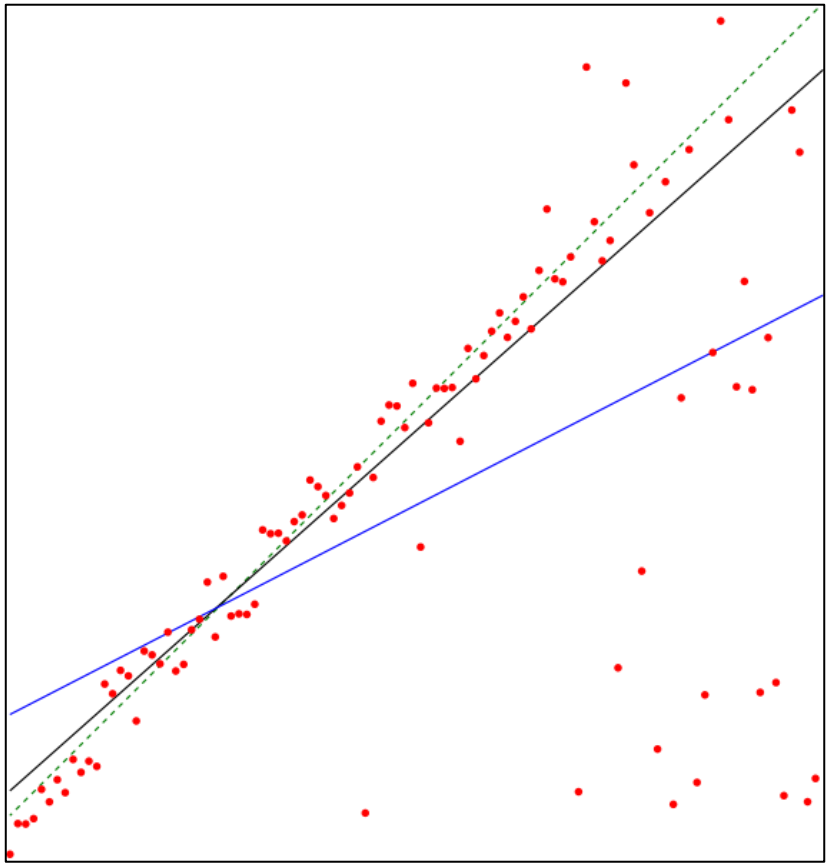


The workflow of Pycoevol

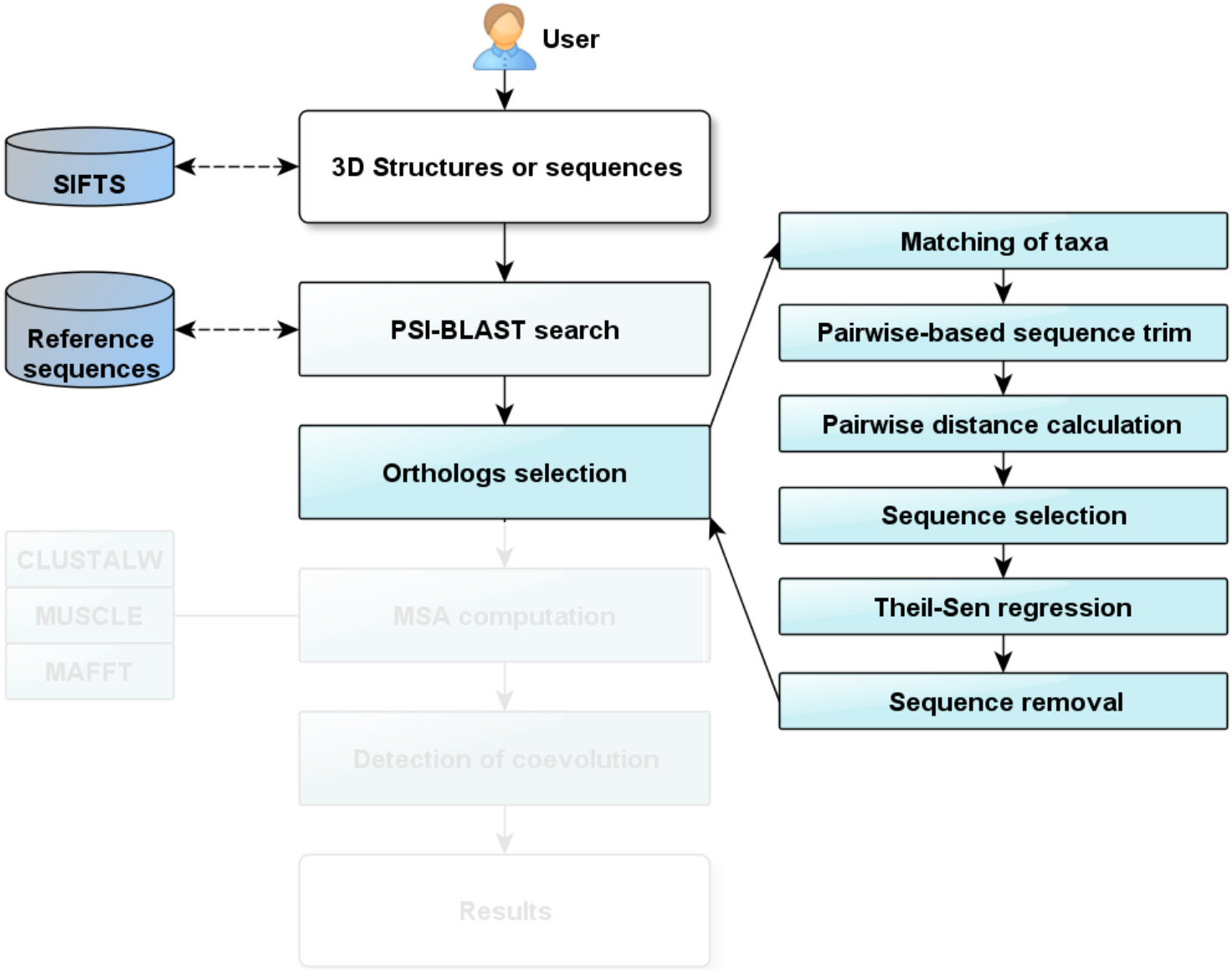


Thiel-Sen regression and sequence removal

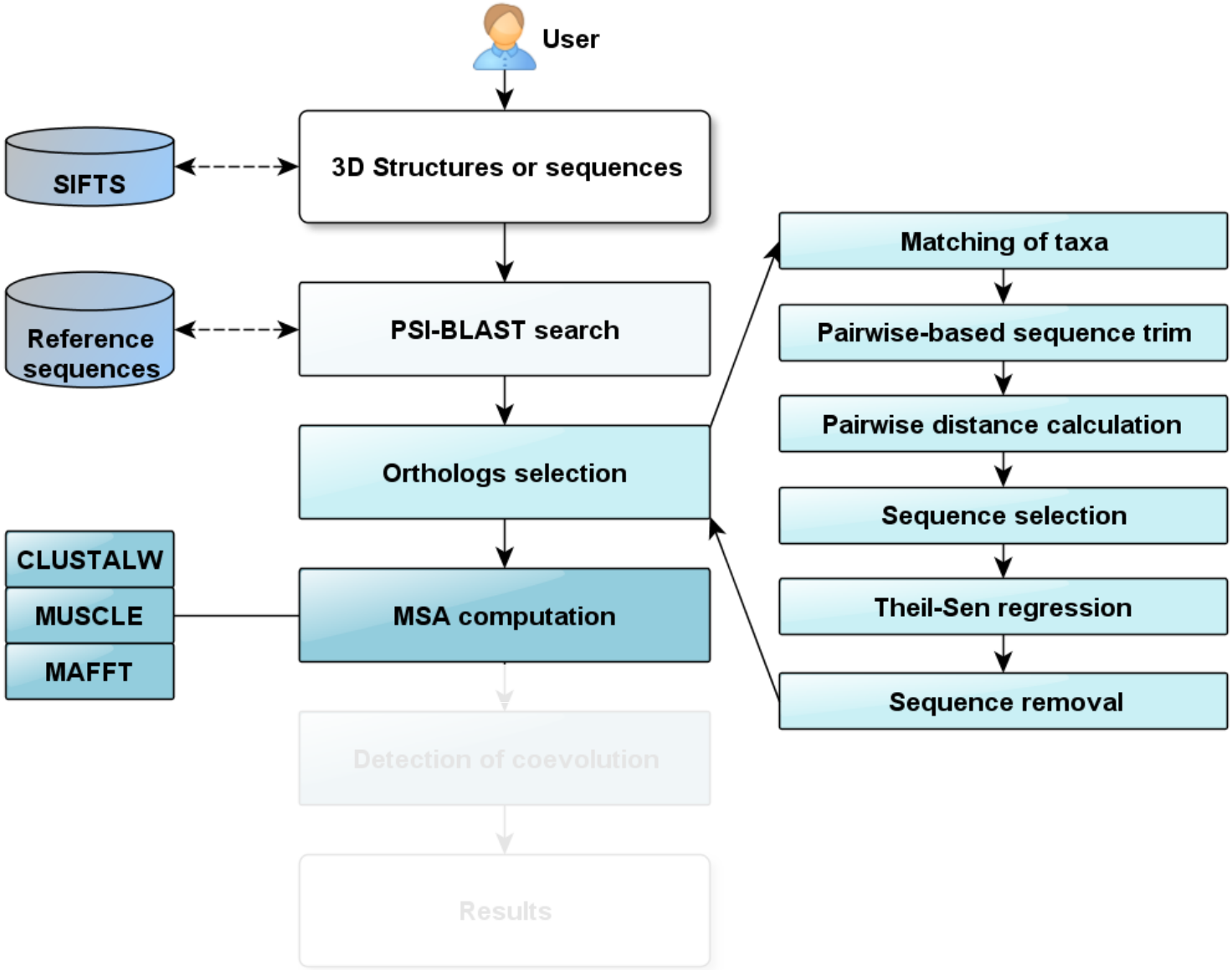
- *Median m of the slopes* $(y_j - y_i)/(x_j - x_i)$ *determined by all pairs of sample points* $\in (x,y)$, *which have distinct x-coordinates [Theil, 1950] and [Sen, 1968]*



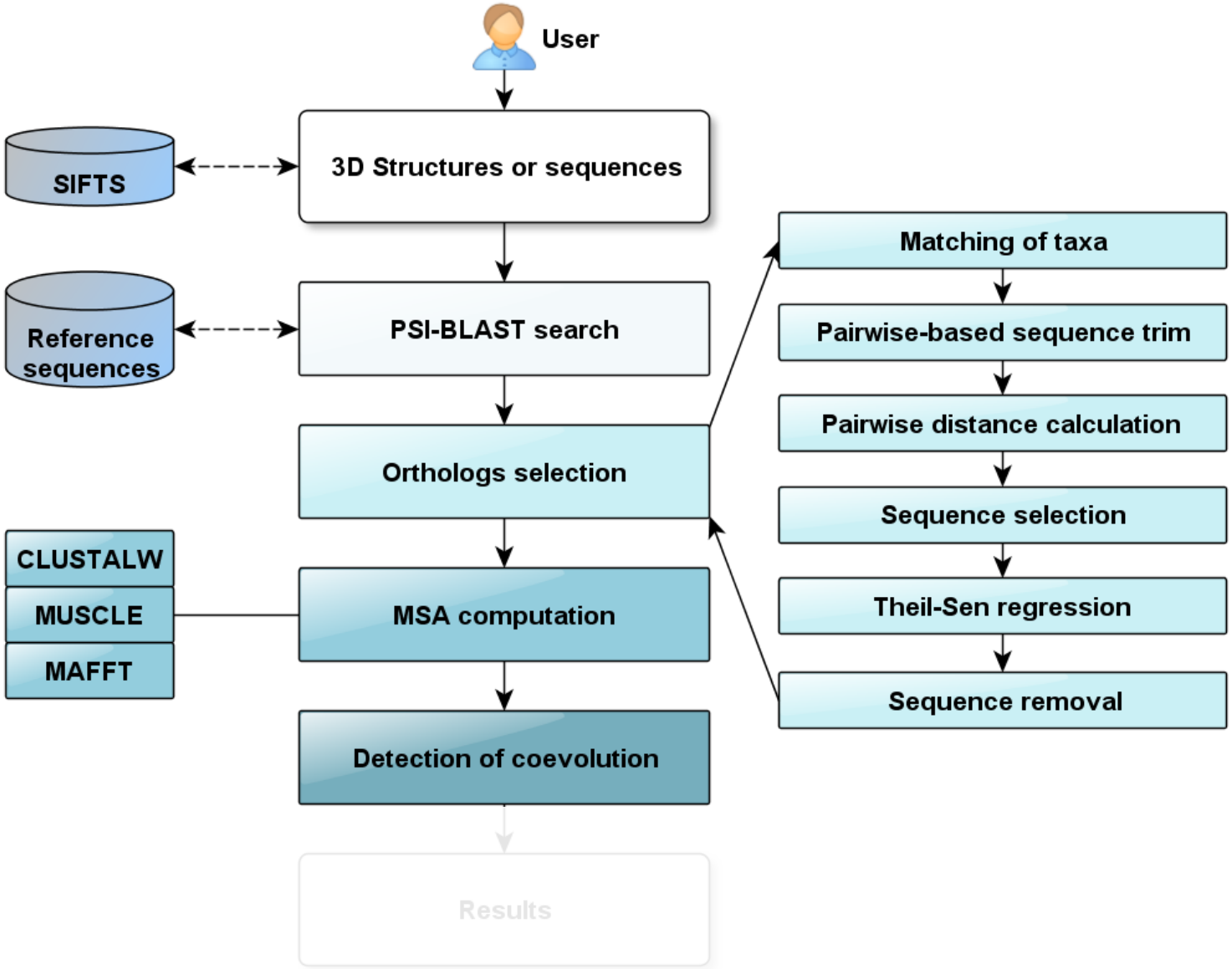
The workflow of Pycoevol



The workflow of Pycoevol



The workflow of Pycoevol



Matrix-based methods

- Contact Preferences, Volume Normalized (**CPVN**) *[Glaser et al, 2001]*
- Contact PDB-derived Likelihood Matrix (**CLM**) *[Singer et al, 2002]*
- Residue-residue Volume Normalized (**VOL**) *[Esque et al, 2010]*

$$CM_{x,y} = \sum_{i}^{n} \sum_{j}^{n} \frac{S_{ij}}{n}$$

Correlation-based methods

- Pearson's correlation (**Pearson**) *[Göbel et al, 1994]*
- Spearman's rank correlation (**Spearman**) *[Pazos et al, 1997]*
- McLachlan Based Substitution Correlation (**McBASC**) *[Fodor and Aldrich, 2004]*
- ***Quartets*** *[Galitsky, 2002]*

$$CM_{x,y} = \frac{1}{N^2} \sum_i \sum_j \frac{W_{ij}(S_{xij} - \langle S_x \rangle)(S_{yij} - \langle S_y \rangle)}{\sigma_x \sigma_y}$$

- Observed Minus Expected Squared (**OMES**) *[Kass and Horovitz, 2002]*

$$CM_{x,y} = \sum_l^L \frac{(N_{obs} - \frac{C_{xi} C_{yj}}{N_{valid}})^2}{N_{valid}}$$

Perturbation-based methods

- Mutual Information (**MI**) *[Gloor et al, 2005]*
- MI by pair Entropy (**MI/E**) *[Martin et al, 2005]*
- Row and Column Weighed MI (**RCW MI**) *[Gouveia-Oliveira et al, 2007]*

$$MI_{x,y} = \sum_i \sum_j P(x_i y_j) log \frac{P(x_i y_j)}{P(x_i)P(y_j)}$$

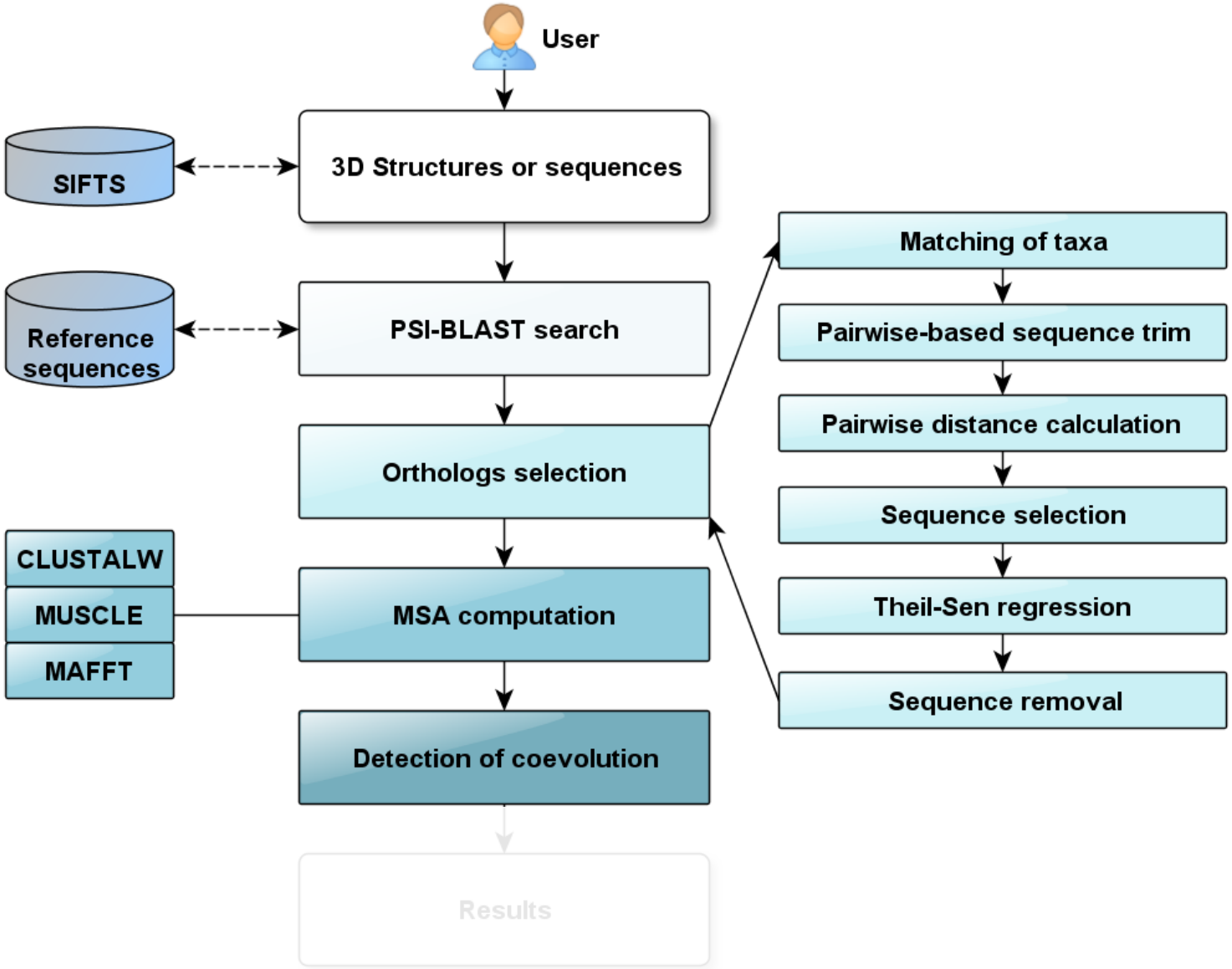
- Statistical Coupling Analysis (**SCA**) *[Lockless and Ranganathan, 1999]*

$$\Delta\Delta G_{x,y} = \sqrt{\sum_i (ln P^i_{x|\delta y} - P^i_x)^2}$$

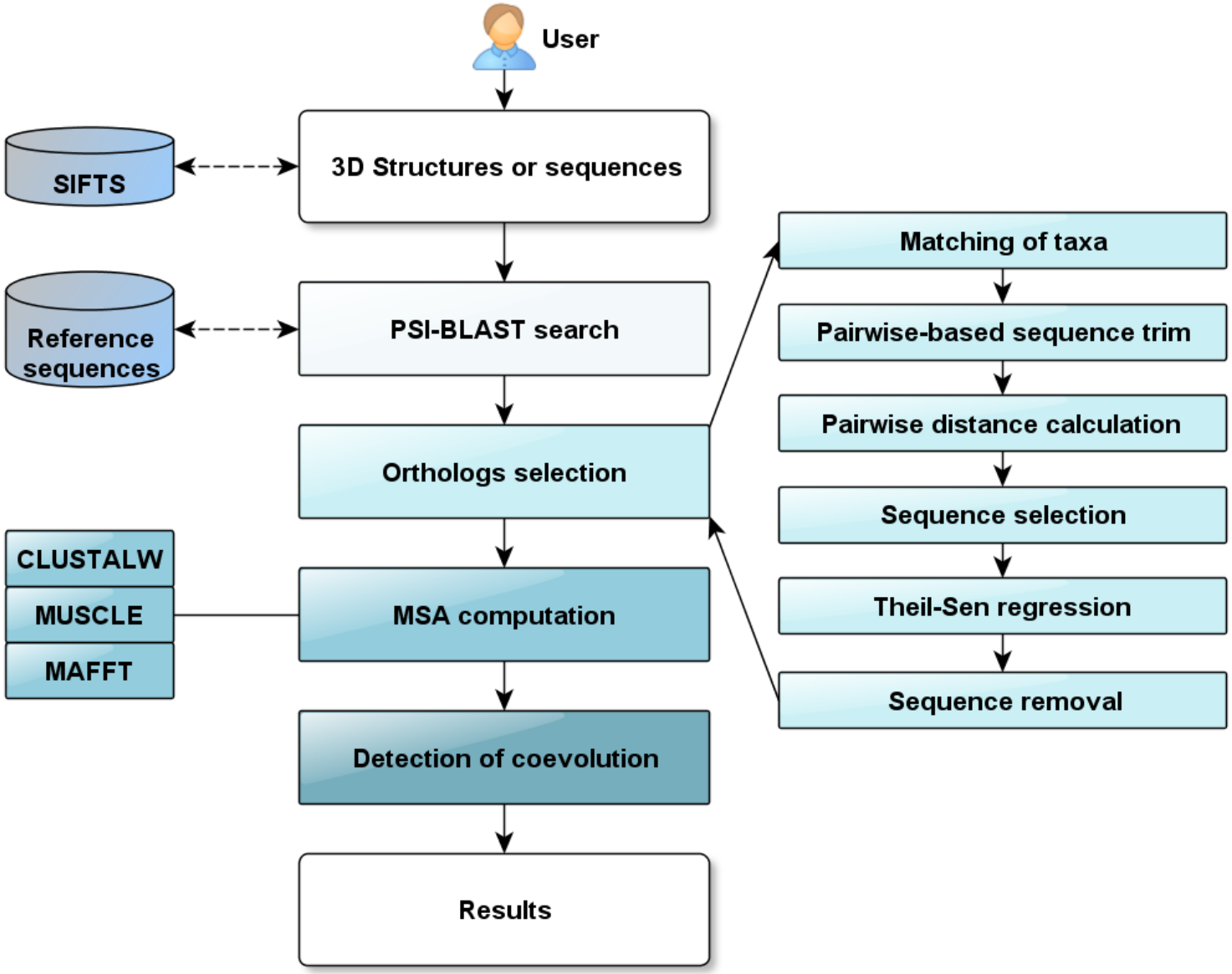
- Explicit Likelihood of Subset Covariation (**ELSC**) *[Dekker et al, 2004]*

$$\Lambda^y_x = \prod_r^{20} \frac{\binom{N_{r,y}}{n_{r,y}}}{\binom{N_{r,y}}{m_{r,y}}}$$

The workflow of Pycoevol



The workflow of Pycoevol



Output

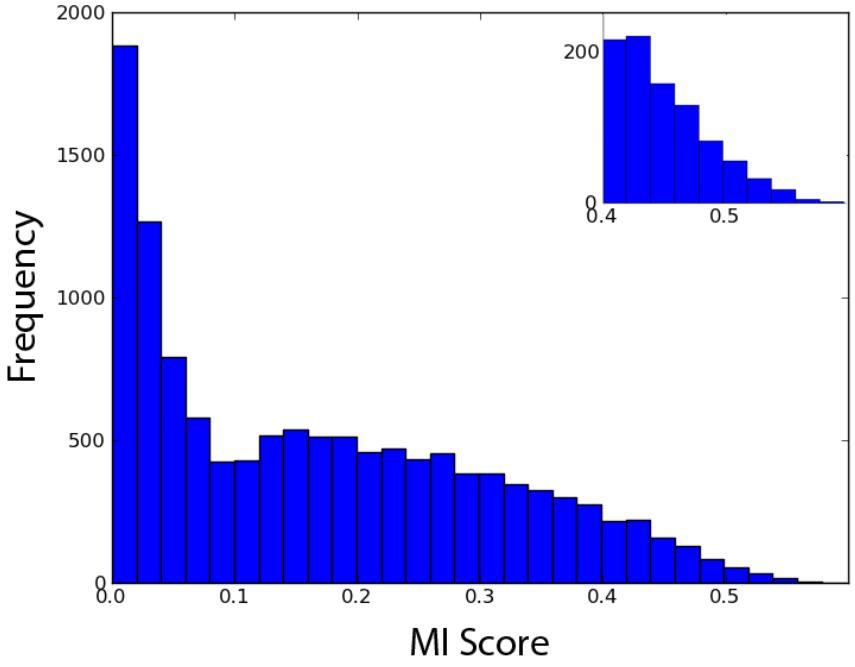
- Matrix of coevolution scores and lists with the most significant coevolving residues

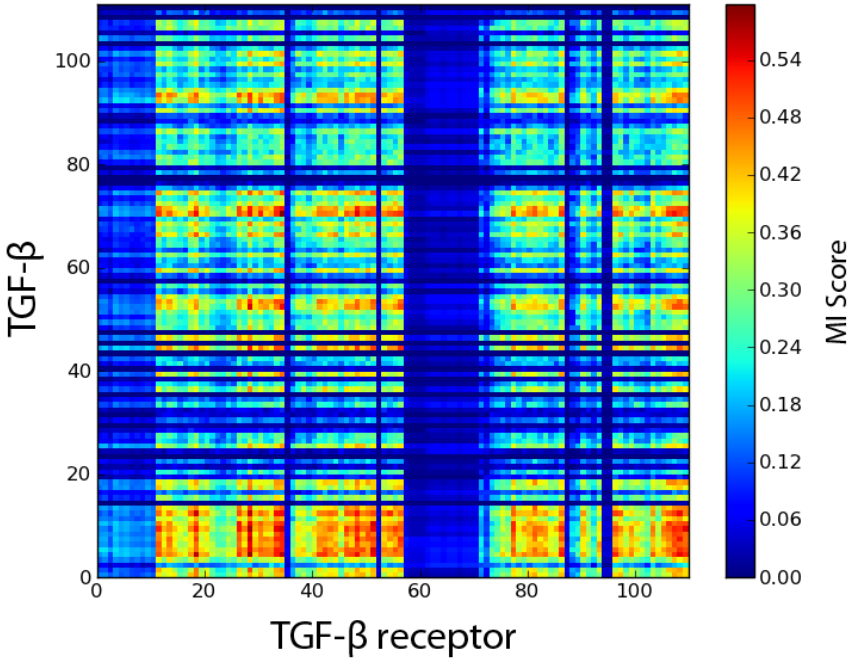
	1	2	3	4	5	6	7	x
1	0.12	0.23	0.11	0.01	0.02	0.21	0.04	...
2	0.09	0.14	0.34	0.09	0.09	0.05	0.05	...
3	0.00	0.20	0.07	0.05	0.06	0.14	0.04	...
4	0.23	0.13	0.07	0.03	0.13	0.23	0.02	...
5	0.04	0.00	0.14	0.21	0.10	0.05	0.01	...
6	0.05	0.23	0.02	0.45	0.12	0.12	0.04	...
7	0.12	0.01	0.01	0.02	0.29	0.30	0.04	...
y	...	...	...	...	...	...	...	...

Protein A	Protein B	Score
13	67	0.61
14	187	0.59
56	87	0.58
79	34	0.56
102	35	0.51
178	35	0.50
x	y	...

Output

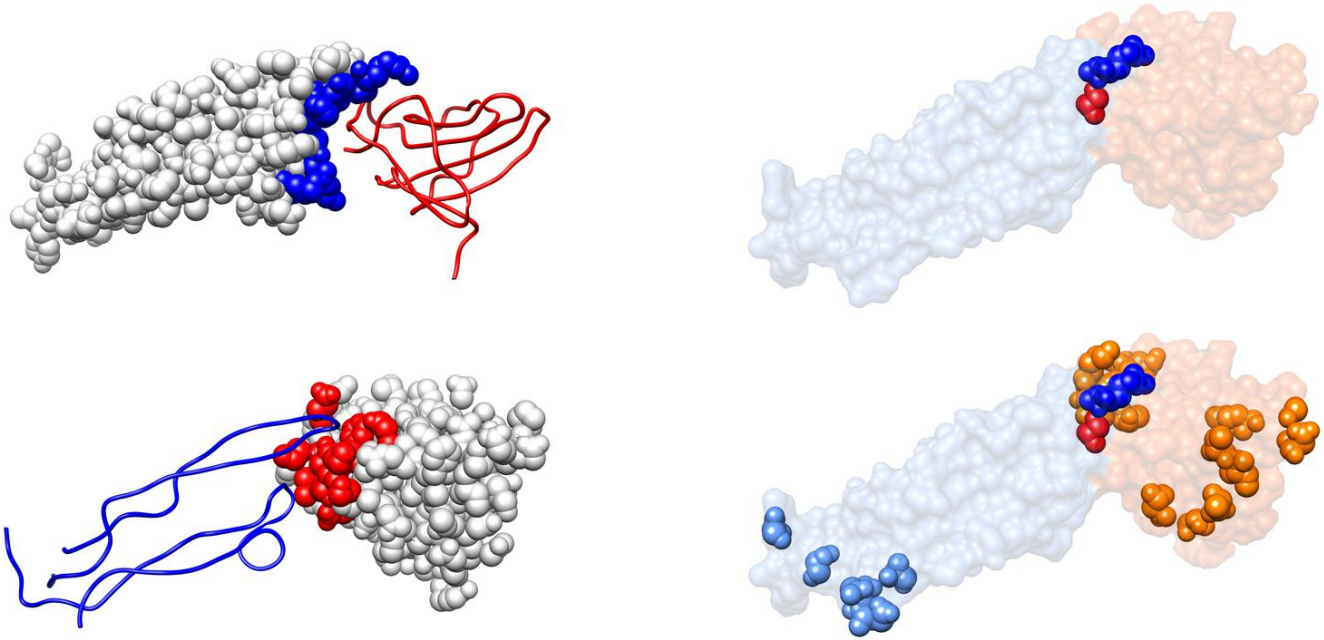
- Histograms and heatmaps





Output

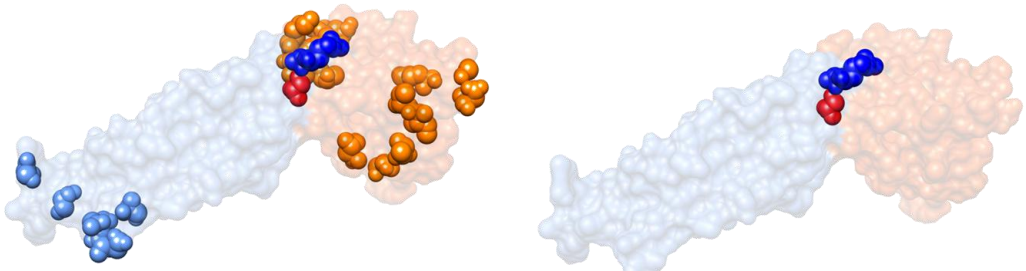
- *Pymol* scripts for rapid visualization of 3D structures

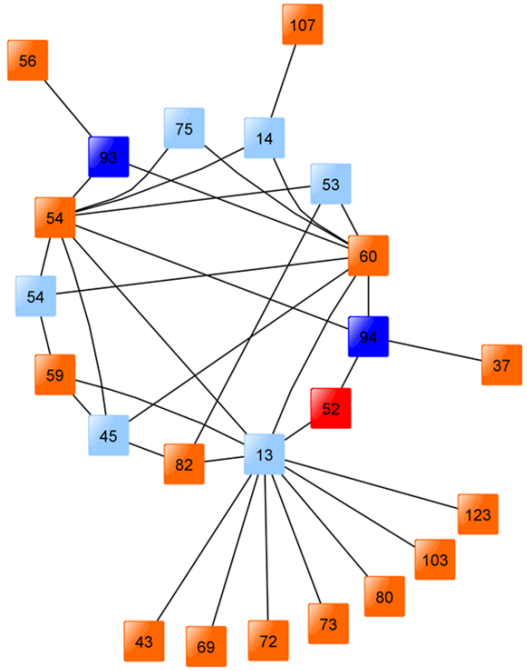


Example

Protein complex TGF-β3/TGF-β receptor type II (1ktz)

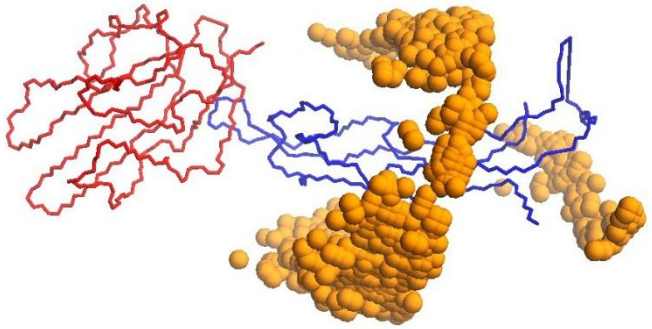
- Most residues were at the surface level
- 3 out of 23 residues were at the interface
- 8 residues from TGF-β3
- 15 residues from TGB-β receptor





Improving protein-protein docking

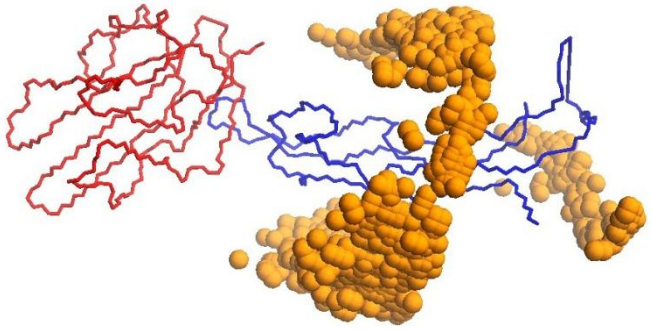
- Possible pairs of contacts = 11772
- Tested contacts = 120
- Reduced the search space to 0.01 % of possible contact points



Difficult complex to model

Improving protein-protein docking

- Possible pairs of contacts = 11772
- Tested contacts = 120
- Reduced the search space to 0.01 % of possible contact points



Difficult complex to model

➢ Given the complexity of protein-protein docking, finding even only one positive interface contact can help constraint the search space and improve the accuracy of constrained docking algorithms such as BiGGER [Palma et al, 2000]

Pycoevol

- Set of tools for the study of inter-protein coevolution and interaction

Pycoevol

- Set of tools for the study of inter-protein coevolution and interaction
- Implements the classic workflow with extended capabilities

Pycoevol

- Set of tools for the study of inter-protein coevolution and interaction
- Implements the classic workflow with extended capabilities
- Automates access to remote databases and third-party applications

Pycoevol

- Set of tools for the study of inter-protein coevolution and interaction
- Implements the classic workflow with extended capabilities
- Automates access to remote databases and third-party applications
- Simplifies the coevolution analysis and the interpretation of results

Pycoevol

- Set of tools for the study of inter-protein coevolution and interaction
- Implements the classic workflow with extended capabilities
- Automates access to remote databases and third-party applications
- Simplifies the coevolution analysis and the interpretation of results
- Implemented in Python and platform independent

Pycoevol

- Set of tools for the study of inter-protein coevolution and interaction
- Implements the classic workflow with extended capabilities
- Automates access to remote databases and third-party applications
- Simplifies the coevolution analysis and the interpretation of results
- Implemented in Python and platform independent
- Open source and public domain

Pycoevol

- Set of tools for the study of inter-protein coevolution and interaction
- Implements the classic workflow with extended capabilities
- Automates access to remote databases and third-party applications
- Simplifies the coevolution analysis and the interpretation of results
- Implemented in Python and platform independent
- Open source and public domain
- Source code at https://github.com/fmadeira/pycoevol

Pycoevol

- Survey which method is better suited for the identification of interface contact points using a large protein complex dataset

- *Thanks for your attention!*

Fábio Madeira - fmadeira@campus.fct.unl.pt

CENTRIA-DI, Faculdade de Ciências e Tecnologia

Universidade Nova de Lisboa