UniProt Proteomes и EMBOSS

UniProt Proteomes

Что такое протеом в UniProt?

В теории: совокупность белков, экспрессирующихся в одном организме.

На практике: совокупность трансляций открытых рамок считывания из полного генома.

Технически: запись в базе данных Proteomes.

- ссылки на записи UniProtKB и/или UniParc
- метаданные (статус протеома, организм, ссылка на сборку генома и т.д.)

Этапы добавления нового протеома

- Добавление новой полногеномной сборки, содержащей информацию о открытых рамках считывания, в нуклеотидный архив.
- Проверка на избыточность.
- Создание записей, оценка качества и полноты.

А дальше может происходить:

- Добавление/удаление белков.
- Перевод протеома в разряд референсных.
- Удаление протеома.

Статусы протеомов в UniProt

Протеомы в UniProt имеют один из статусов, перечисленных ниже.

Референсные (reference) – вручную или автоматически отобранные в качестве лучшего среди доступных протеомов таксономической группы (обычно вида).

Избыточные (redundant) – слишком сильно похожие на другой протеом; для белков из таких протеомов не создаются записи в UniProtKB, только в UniParc.

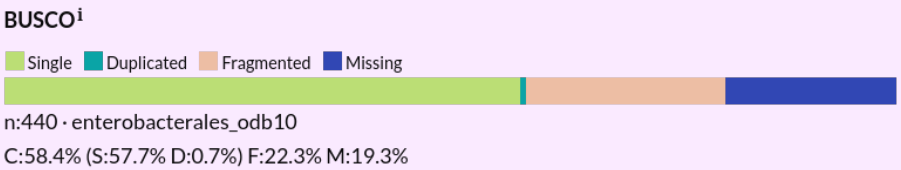
Удаленные (excluded) – протеомы, удаленные вслед за геномной сборкой из RefSeq; белки из таких протеомов удаляются из TrEMBL.

Прочие (other) – все остальные, обычные протеомы.

Меры качества и полноты

CPD (Complete Proteome Detector) – сравнение с протеомами близких организмов на предмет отличия в размерах. По результатам присваивают категорию: Standard, Close to standard (high value), Close to standard (low value), Outlier (high value), Outlier (low value) или Unknown.

BUSCO (Benchmarking Universal Single-Copy Ortholog) – внешний алгоритм оценки качества по наличию представителей референсных ортологичных групп белков. Каждой группе ортологов присваивается один из 4 статусов: Single, Duplicated, Fragmented, Missing. Результат – процент групп из каждой категории в графическом или числовом представлении.



Пан-протеомы в UniProt

Пан-протеом (Pan proteome) – совокупность разных белков из группы близкородственных организмов.

Включает в себя:

- все белки из референсного протеома;
- по одному представителю из всех кластеров UniRef50 нереференсных протеомов, которые не содержат белков из референсного.

Пан-протеомы не выделены в отдельную базу, но и не имеют своих записей в базе Proteomes. Идентификатор пан-протеома совпадает с ID референсного протеома, входящего в его состав.

EMBOSS

EMBOSS

European **M**olecular **B**iology **O**pen **S**oftware **S**uite

Пакет консольных биоинформатических программ.

- унифицированный интерфейс
- общий формат для задания адреса последовательностей (USA)
- есть программы для повседневных задач, возникающих при работе с биологическими последовательностями
- пакет перестал развиваться в 2013, программы устаревают

Помощь по программам

Можно получить справку в командной строке:

```
Краткое описание основных опций:
kodomo:~$ программа-из-emboss -help

Описание всех имеющихся опций:
kodomo:~$ программа-из-emboss -help -verbose

Подробное описание программы:
kodomo:~$ tfm 'программа-из-emboss'

Поиск программы по описанию:
kodomo:~$ wossname 'alignment'

У всех программ есть man, по объему это примерно -help
kodomo:~$ man 'программа-из-emboss'
```

Или можно читать описания в интернете:

http://emboss.open-bio.org/ путаный официальный сайт

http://emboss.sourceforge.net/ лучше организован, но бывают проблемы с доступом

Унифицированный адрес последовательности (USA)

Uniform **S**equence **A**ddress

```
DB:entry[start:end:reverse]
format::file:entry[start:end:reverse]
@listfile
```

Все варианты USA описаны здесь:
http://emboss.sourceforge.net/docs/themes/UniformSequenceAddress.html

Список поддерживаемых форматов файлов доступен здесь:
http://emboss.sourceforge.net/docs/themes/SequenceFormats.html

Список баз данных можно узнать с помощью команды `showdb` . На kodomo есть локальная копия Swiss-Prot, и настроено скачивание одиночных записей из ENA/DDBJ и UniProtKB.

В именах файлов и записей можно использовать маски (только `*` и `?`).
Не забывайте про экранирование!

Аргументы командной строки

- аргументы называются qualifiers
- бывают пяти типов: standard, additional, advanced, associated и general
- всегда задаются в виде опций, начинающихся с *одного* символа -
- название опции можно сокращать, пока понятно, какая опция имеется в виду
- нельзя склеивать названия нескольких опций после одного -
- почти все опции требуют один аргумент
- у опций типа boolean аргумент можно опускать, имея в виду значение Y

Standard qualifiers

Обязательные аргументы

- если не заданы, будут запрошены с `STDIN` в процессе исполнения
- иногда могут задаваться в виде позиционных аргументов (т.е. без указания названия опции), в этом случае название опции заключено в [] на странице `-help`
- иногда для них есть значение по умолчанию, которое можно активировать опцией `-auto`

Пример:

```
kodomo:~$ infoseq -sequence 'seq.fasta'
или (то же самое):
kodomo:~$ infoseq 'seq.fasta'
```

Additional qualifiers

Дополнительные аргументы

- если не заданы, будут использованы значения по умолчанию (будут запрошены с `STDIN` в интерактивном режиме, если задана опция `-options`)
- значения по умолчанию указаны в [] на странице `-help`

Пример:

```
kodomo:~$ infoseq seq.fasta -outfile 'report.txt'
```

Advanced qualifiers

Расширенные аргументы

- предполагается, что они редко потребуются рядовым пользователям
- отображаются на странице `-help` без опции `-verbose`

Пример:

```
kodomo:~$ infoseq seq.fasta -delimiter ';'
```

Associated qualifiers

Ассоциированные аргументы

- уточняют значения других аргументов
- не отображаются на странице `-help` без опции `-verbose`
- на странице `-help -verbose` указано, какой аргумент они уточняют

Пример:

```
kodomo:~$ infoseq seq.fasta -squick 'Y'
```

General qualifiers

Общие аргументы

- есть у всех программ EMBOSS
- не отображаются на странице `-help` без опции `-verbose` (за исключением самой опции `-help`)
- служат либо для получения служебной информации о программе, либо для переключения режима взаимодействия с программой

Пример:

```
kodomo:~$ infoseq -help 'Y' -verbose 'N'
```

Использование в конвейерах

Программы пакета EMBOSS неудобно использовать в конвейерах, так как они:

- используют файловый ввод/вывод (а не стандартные потоки);
- переключаются в интерактивный режим в случае указания не всех обязательных аргументов (даже при наличии подходящих умолчательных значений);
- выводят бесполезные информационные сообщения.

Использование в конвейерах

Есть общие (general) опции, позволяющие решить некоторые или все проблемы:

-auto – использовать умолчательные значения даже для пропущенных обязательных аргументов (+ отключить информационные сообщения);

-filter – заменить умолчательные ввод и вывод на стандартные потоки, и еще все то, что делает `-auto` .

Советую *всегда* использовать `-filter` , не могу придумать ситуацию, когда эта опция помешает.

Проблемы с выводом сообщений

Все информационные сообщения, в том числе `-help`, программы EMBOSS выводят в `STDERR`, а не в `STDOUT`.

```
Слить STDOUT и STDERR и перенаправить в файл:
kodomo:~$ seqret -help &> 'seqret_help.txt'

Слить STDOUT и STDERR и передать следующей команде:
kodomo:~$ seqret -help -verbose |& less

Убить STDERR (перенаправить в черную дыру):
kodomo:~$ seqret 'seqs.fasta' 'plain::stdout' 2> '/dev/null' | less

Отключить сообщения на уровне команды EMBOSS:
kodomo:~$ seqret -filter 'seqs.fasta' | less
или
kodomo:~$ seqret -auto 'seqs.fasta' 'out.fasta'
```

Программа seqret

от **seq**uence **ret**urn

Получает последовательности согласно USA из первого аргумента и записывает их согласно USA из второго аргумента (можно указать только имя файла и его формат).

```
Изменить формат файла:
kodomo:~$ seqret -filter 'seqs.fasta' 'asn1::seqs.txt'

Вырезать участок последовательности:
kodomo:~$ seqret -filter 'seqs.fasta:*[1:5]' 'first_aa.fasta'

Отобрать последовательности по маске имени:
kodomo:~$ seqret -filter 'seqs.fasta:abc*' 'ncbi::abc.txt'

Объединить разные последовательности в один файл:
kodomo:~$ cat 'usa.list'
seqs.fasta:qwe123[1:30]
seqs.fasta:qwe123[31:60]
ncbi::abc.txt
kodomo:~$ seqret -filter '@usa.list' 'ncbi::all.fasta'
```

Программа entret

от **ent**ry **ret**urn

Нужна для скачивания записей из баз данных.

```
Получить последовательность из базы можно с помощью seqret:
kodomo:~$ seqret -filter 'sw:ENO_ECOLI' 'eno_ecoli.fasta'

Можно даже сохранить её в нативном формате:
kodomo:~$ seqret -filter 'sw:ENO_ECOLI' 'swiss::eno_ecoli.seqret'

Но seqret не гарантирует сохранение всех аннотаций!
kodomo:~$ grep -c '^FT' 'eno_ecoli.seqret'
0

Скачивать полные записи следует с помощью entret:
kodomo:~$ entret -filter 'sw:ENO_ECOLI' 'eno_ecoli.entret'
kodomo:~$ grep -c '^FT' 'eno_ecoli.entret'
151
```

Второй аргумент entret – **не** USA, а просто имя файла.