In-silico cleavage of polypeptides using the cleaver package

Sebastian Gibb[*]

May 15, 2016

Contents

1 Introduction

Most proteomics experiments need protein (peptide) separation and cleavage procedures before these molecules could be analyzed or identified by mass spectrometry or other analytical tools.
cleaver allows in-silico cleavage of polypeptide sequences to e.g. create theoretical mass spectrometry data.
The cleavage rules are taken from the ExPASy PeptideCutter tool[?] .

2 Simple Usage

Loading the *cleaver* package:

```
> library("cleaver")
```

Getting help and list all available cleavage rules:

```
> help("cleave")
```

Cleaving of *Gastric juice peptide 1 (P01358)* using *Trypsin*:

```
> ## cleave it
> cleave("LAAGKVEDSD", enzym="trypsin")

$LAAGKVEDSD
[1] "LAAGK" "VEDSD"
```

[*] mail@sebastiangibb.de

```
> ## get the cleavage ranges
> cleavageRanges("LAAGKVEDSD", enzym="trypsin")

$LAAGKVEDSD
     start end
[1,]     1   5
[2,]     6  10

> ## get only cleavage sites
> cleavageSites("LAAGKVEDSD", enzym="trypsin")

$LAAGKVEDSD
[1] 5
```

Sometimes cleavage is not perfect and the enzym miss some cleavage positions:

```
> ## miss one cleavage position
> cleave("LAAGKVEDSD", enzym="trypsin", missedCleavages=1)

$LAAGKVEDSD
[1] "LAAGKVEDSD"

> cleavageRanges("LAAGKVEDSD", enzym="trypsin", missedCleavages=1)

$LAAGKVEDSD
     start end
[1,]     1  10

> ## miss zero or one cleavage positions
> cleave("LAAGKVEDSD", enzym="trypsin", missedCleavages=0:1)

$LAAGKVEDSD
[1] "LAAGK"      "VEDSD"      "LAAGKVEDSD"

> cleavageRanges("LAAGKVEDSD", enzym="trypsin", missedCleavages=0:1)

$LAAGKVEDSD
     start end
[1,]     1   5
[2,]     6  10
[3,]     1  10
```

Combine *cleaver* and the *Biostrings* R package[?] :

```
> ## create AAStringSet object
> p <- AAStringSet(c(gaju="LAAGKVEDSD", pnm="AGEPKLDAGV"))
>
> ## cleave it
> cleave(p, enzym="trypsin")

AAStringSetList of length 2
[["gaju"]] LAAGK VEDSD
[["pnm"]] AGEPK LDAGV

> cleavageRanges(p, enzym="trypsin")
```

```
IRangesList of length 2
$gaju
IRanges object with 2 ranges and 0 metadata columns:
          start       end     width
      <integer> <integer> <integer>
  [1]         1         5         5
  [2]         6        10         5

$pnm
IRanges object with 2 ranges and 0 metadata columns:
          start       end     width
      <integer> <integer> <integer>
  [1]         1         5         5
  [2]         6        10         5

> cleavageSites(p, enzym="trypsin")

$gaju
[1] 5

$pnm
[1] 5
```

3 Insulin & Somatostatin Example

Downloading *Insulin (P01308)* and *Somatostatin (P61278)* sequences from the UniProt[?] database using the *UniProt.ws* R package[?] .

```
> ## load UniProt.ws library
> library("UniProt.ws")
>
> ## select species Homo sapiens
> UniProt.ws <- UniProt.ws(taxId=9606)
>
> ## download sequences of Insulin/Somatostatin
> s <- select(UniProt.ws, keys=c("P01308", "P61278"), columns=c("SEQUENCE"))

Getting extra data for P01308,P61278

'select()' returned 1:1 mapping between keys and columns

> ## fetch only sequences
> sequences <- setNames(s$SEQUENCE, s$UNIPROTKB)
>
> ## remove whitespaces
> sequences <- gsub(pattern="[[:space:]]", replacement="", x=sequences)
```

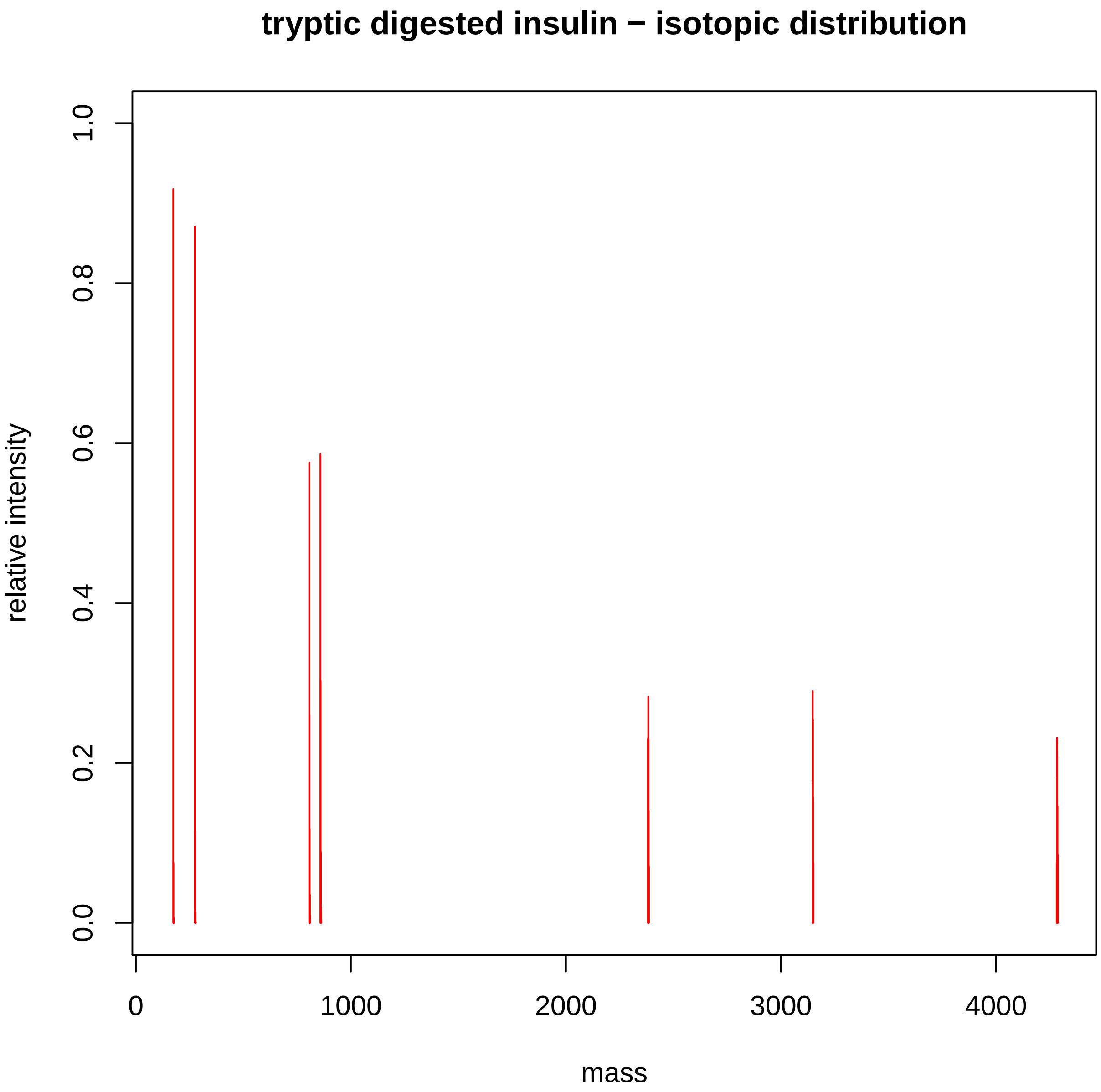
Cleaving using *Pepsin*:

```
> cleave(sequences, enzym="pepsin")

$P01308
 [1] "MA"              "L"               "WMRLLP"          "LL"
 [5] "A"               "WGPDPAAA"        "F"               "VNQH"
 [9] "CGSH"            "VEA"             "Y"               "VCGERG"
[13] "FF"              "YTPKTRREAED"     "QVGQVE"          "GGGPGAGS"
[17] "LQP"             "LA"              "EGS"             "QKRGIVEQCCTSICS"
[21] "YQ"              "ENYCN"

$P61278
 [1] "ML"                    "SCRL"                  "QCA"
 [4] "L"                     "AA"                    "SIV"
 [7] "A"                     "GCVTGAPSDPRL"          "RQ"
[10] "FL"                    "QKS"                   "LAAAAGKQEL"
[13] "AKY"                   "AE"                    "SEPNQTENDA"
[16] "LEPED"                 "SQAAEQDEMRL"           "EL"
[19] "QRSANSNPAMAPRERKAGCKN" "FF"                    "WKT"
[22] "FTSC"
```

4 Isotopic Distribution Of Tryptic Digested Insulin

A common use case of in-silico cleavage is the calculation of the isotopic distribution of peptides (which were enzymatic digested in the in-vitro experimental workflow). Here the *BRAIN* R package[? ?] is used to calculate the isotopic distribution of *cleaver*'s output. (please note: it is only a toy example, e.g. the relation of intensity values between peptides isn't correct).

```
> ## load BRAIN library
> library("BRAIN")
>
> ## cleave insulin
> cleavedInsulin <- cleave(sequences[1], enzym="trypsin")[[1]]
>
> ## create empty plot area
> plot(NA, xlim=c(150, 4300), ylim=c(0, 1),
+      xlab="mass", ylab="relative intensity",
+      main="tryptic digested insulin - isotopic distribution")
>
> ## loop through peptides
> for (i in seq(along=cleavedInsulin)) {
+   ## count C, H, N, O, S atoms in current peptide
+   atoms <- BRAIN::getAtomsFromSeq(cleavedInsulin[[i]])
+   ## calculate isotopic distribution
+   d <- useBRAIN(atoms)
+   ## draw peaks
+   lines(d$masses, d$isoDistr, type="h", col=2)
+ }
```



5 Session Information

- R version 3.3.0 (2016-05-03), `x86_64-apple-darwin13.4.0`
- Locale: `C/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8`
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, stats4, utils
- Other packages: BRAIN 1.18.0, BiocGenerics 0.18.0, Biostrings 2.40.0, DBI 0.4-1, IRanges 2.6.0, PolynomF 0.94, RCurl 1.95-4.8, RSQLite 1.0.0, S4Vectors 0.10.0, UniProt.ws 2.12.0, XVector 0.12.0, bitops 1.0-6, cleaver 1.10.2, knitr 1.13, lattice 0.20-33
- Loaded via a namespace (and not attached): AnnotationDbi 1.34.2, Biobase 2.32.0, BiocStyle 2.0.2, evaluate 0.9, formatR 1.4, grid 3.3.0, highr 0.6, magrittr 1.5, stringi 1.0-1, stringr 1.0.0, tools 3.3.0, zlibbioc 1.18.0