In-silico cleavage of polypeptides using the cleaver package

Sebastian Gibb*

April 24, 2017

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[4].

2 Simple Usage

Loading the *cleaver* package:

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

Getting help and list all available cleavage rules:

*mail@sebastiangibb.de

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

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

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

$LAAGKVEDSD
[1] "LAAGK" "VEDSD"

> ## 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[5]:

```
> ## 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[6] database using the *UniProt.ws* R package[1].

```
> ## 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")
```

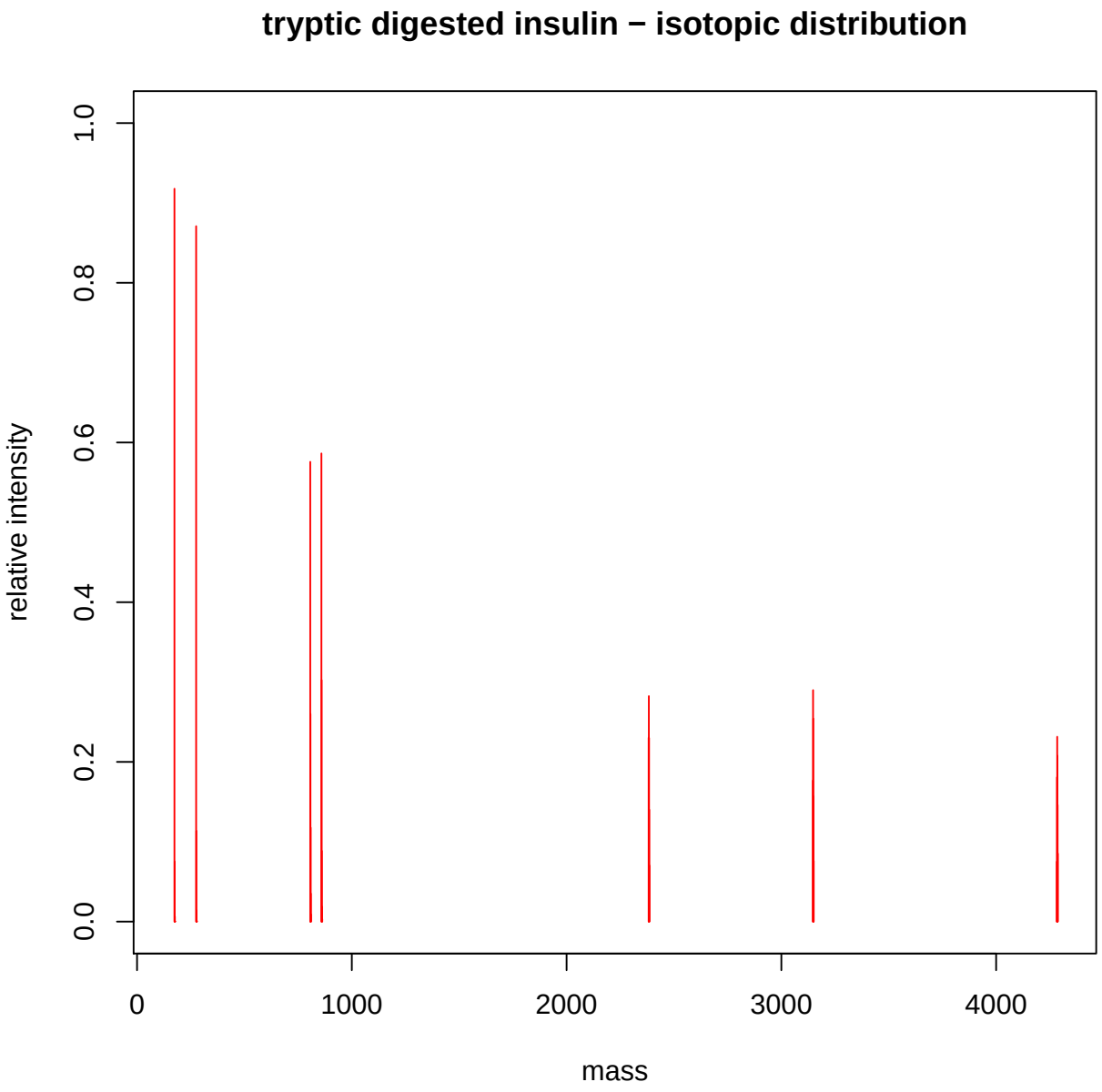
```
$P01308
 [1] "MA"              "L"               "WMRLLP"          "LL"
 [5] "A"               "WGPDPAAA"        "F"               "VNQH"
 [9] "CGSH"            "VEA"             "Y"               "VCGERG"
[13] "FF"              "YTPKTRREAED"     "QVGQVE"          "GGGPGAGS"
[17] "LQP"             "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[2;3] 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 3.4.0 (2017-04-21), x86_64-pc-linux-gnu
- Locale: LC_CTYPE=en_US.UTF-8, LC_NUMERIC=C, LC_TIME=en_US.UTF-8, LC_COLLATE=C,

LC_MONETARY=en_US.UTF-8, LC_MESSAGES=en_US.UTF-8, LC_PAPER=en_US.UTF-8, LC_NAME=C, LC_ADDRESS=C, LC_TELEPHONE=C, LC_MEASUREMENT=en_US.UTF-8, LC_IDENTIFICATION=C

- Running under: Ubuntu 16.04.2 LTS
- Matrix products: default
- BLAS: /home/biocbuild/bbs-3.5-bioc/R/lib/libRblas.so
- LAPACK: /home/biocbuild/bbs-3.5-bioc/R/lib/libRlapack.so
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, stats4, utils
- Other packages: BRAIN 1.22.0, BiocGenerics 0.22.0, Biostrings 2.44.0, IRanges 2.10.0, PolynomF 0.94, RCurl 1.95-4.8, RSQLite 1.1-2, S4Vectors 0.14.0, UniProt.ws 2.16.0, XVector 0.16.0, bitops 1.0-6, cleaver 1.14.0, knitr 1.15.1, lattice 0.20-35
- Loaded via a namespace (and not attached): AnnotationDbi 1.38.0, Biobase 2.36.0, BiocStyle 2.4.0, DBI 0.6-1, Rcpp 0.12.10, backports 1.0.5, compiler 3.4.0, digest 0.6.12, evaluate 0.10, grid 3.4.0, highr 0.6, htmltools 0.3.5, magrittr 1.5, memoise 1.1.0, rmarkdown 1.4, rprojroot 1.2, stringi 1.1.5, stringr 1.2.0, tools 3.4.0, yaml 2.1.14, zlibbioc 1.22.0

References

[1] Marc Carlson. *UniProt.ws: R Interface to UniProt Web Services*. R package version 2.0.0.

[2] Jürgen Claesen, Piotr Dittwald, Tomasz Burzykowski, and Dirk Valkenborg. An efficient method to calculate the aggregated isotopic distribution and exact center-masses. *Journal of The American Society for Mass Spectrometry*, 23(4):753–763, 2012.

[3] Piotr Dittwald, Jürgen Claesen, Tomasz Burzykowski, Dirk Valkenborg, and Anna Gambin. Brain: A universal tool for high-throughput calculations of the isotopic distribution for mass spectrometry. *Analytical chemistry*, 85(4):1991–1994, 2013.

[4] Elisabeth Gasteiger, Christine Hoogland, Alexandre Gattiker, S'everine Duvaud, Marc R. Wilkins, Ron D. Appel, and Amos Bairoch. Protein identification and analysis tools on the expasy server. In John M. Walker, editor, *The Proteomics Protocols Handbook*, pages 571–607. Humana Press, 2005. ISBN 978-1-58829-343-5. doi: 10.1385/1-59259-890-0:571. URL http://dx.doi.org/10.1385/1-59259-890-0%3A571.

[5] H. Pages, P. Aboyoun, R. Gentleman, and S. DebRoy. *Biostrings: String objects representing biological sequences, and matching algorithms*. R package version 2.28.0.

[6] The UniProt Consortium. Reorganizing the protein space at the universal protein resource (uniprot). *Nucleic Acids Research*, 40(D1):D71–D75, 2012. doi: 10.1093/nar/gkr981. URL http://nar.oxfordjournals.org/content/40/D1/D71.abstract.