**Übungsblatt 2**

**Sequenz Alignment**

**Aufgabe 2.1** A A K M W V

A S K M V V

**2.1.1**

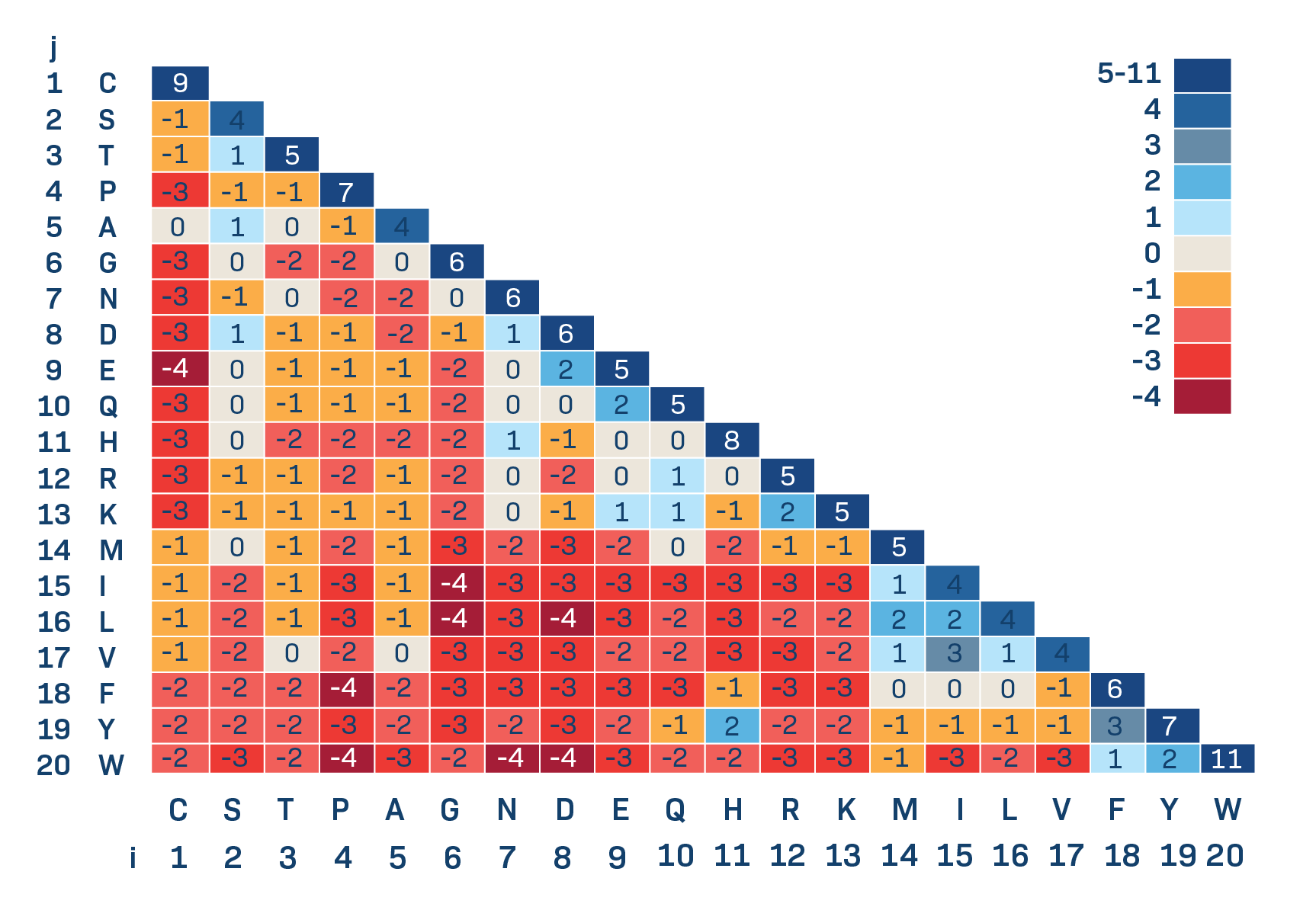
I = 100\*M/L

I = 100\*4/6

I = 66,7% (percent identity)

Bei den Sequenzen handelt es sich wahrscheinlich um zwei homologe Proteine, da I < 30% ist.

**2.1.2**



A A K M W V

A S K M V V

4+1 +5 +5 -3 +4 = 16 (alignment score)

**Aufgabe 2.2**

> setwd("C:/Users/raliz/Documents/Universität/12. Semester/Bioinformatik/R/Übungen/Übung 2/R")

> read.table("alignedSequencesCalmodulin.txt")

V1

1 MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADGNGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDKDGNGYISAAELRHVMTNLGEKLTDEEVDEMIREADIDGDGQVNYEEFVQMMTAK\*

2 MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVDRSLGQNPTEAELQDMINEVDADGNGTIDFPEFLTMMERKMKDTDSEEEIREAFRVFDKDGNGYISAAKLRHVMTNLGEKLTDEEVDEMIREADAAGDGQVNYEEFVQMMTAK\*

> install.packages("stringr")

WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/

Installing package into ‘C:/Users/raliz/AppData/Local/R/win-library/4.2’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/stringr\_1.4.0.zip'

Content type 'application/zip' length 211205 bytes (206 KB)

downloaded 206 KB

package ‘stringr’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\raliz\AppData\Local\Temp\RtmpyeTqFy\downloaded\_packages

> library(stringr)

> str\_extract\_all("MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADGNGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDKDGNGYISAAELRHVMTNLGEKLTDEEVDEMIREADIDGDGQVNYEEFVQMMTAK", boundary("character"))

[[1]]

[1] "M" "A" "D" "Q" "L" "T" "E" "E" "Q" "I" "A" "E" "F" "K" "E" "A" "F" "S" "L" "F" "D" "K" "D" "G" "D" "G" "T" "I" "T" "T" "K" "E" "L"

[34] "G" "T" "V" "M" "R" "S" "L" "G" "Q" "N" "P" "T" "E" "A" "E" "L" "Q" "D" "M" "I" "N" "E" "V" "D" "A" "D" "G" "N" "G" "T" "I" "D" "F"

[67] "P" "E" "F" "L" "T" "M" "M" "A" "R" "K" "M" "K" "D" "T" "D" "S" "E" "E" "E" "I" "R" "E" "A" "F" "R" "V" "F" "D" "K" "D" "G" "N" "G"

[100] "Y" "I" "S" "A" "A" "E" "L" "R" "H" "V" "M" "T" "N" "L" "G" "E" "K" "L" "T" "D" "E" "E" "V" "D" "E" "M" "I" "R" "E" "A" "D" "I" "D"

[133] "G" "D" "G" "Q" "V" "N" "Y" "E" "E" "F" "V" "Q" "M" "M" "T" "A" "K"

> seq1 <- str\_extract\_all("MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADGNGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDKDGNGYISAAELRHVMTNLGEKLTDEEVDEMIREADIDGDGQVNYEEFVQMMTAK", boundary("character"))

> seq2 <- str\_extract\_all("MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVDRSLGQNPTEAELQDMINEVDADGNGTIDFPEFLTMMERKMKDTDSEEEIREAFRVFDKDGNGYISAAKLRHVMTNLGEKLTDEEVDEMIREADAAGDGQVNYEEFVQMMTAK", boundary("character"))

unlist(seq1)

[1] "M" "A" "D" "Q" "L" "T" "E" "E" "Q" "I" "A" "E" "F" "K" "E" "A" "F" "S" "L" "F" "D" "K" "D" "G" "D" "G" "T" "I" "T" "T" "K" "E" "L"

[34] "G" "T" "V" "M" "R" "S" "L" "G" "Q" "N" "P" "T" "E" "A" "E" "L" "Q" "D" "M" "I" "N" "E" "V" "D" "A" "D" "G" "N" "G" "T" "I" "D" "F"

[67] "P" "E" "F" "L" "T" "M" "M" "A" "R" "K" "M" "K" "D" "T" "D" "S" "E" "E" "E" "I" "R" "E" "A" "F" "R" "V" "F" "D" "K" "D" "G" "N" "G"

[100] "Y" "I" "S" "A" "A" "E" "L" "R" "H" "V" "M" "T" "N" "L" "G" "E" "K" "L" "T" "D" "E" "E" "V" "D" "E" "M" "I" "R" "E" "A" "D" "I" "D"

[133] "G" "D" "G" "Q" "V" "N" "Y" "E" "E" "F" "V" "Q" "M" "M" "T" "A" "K"

> unlist(seq2)

[1] "M" "A" "D" "Q" "L" "T" "E" "E" "Q" "I" "A" "E" "F" "K" "E" "A" "F" "S" "L" "F" "D" "K" "D" "G" "D" "G" "T" "I" "T" "T" "K" "E" "L"

[34] "G" "T" "V" "D" "R" "S" "L" "G" "Q" "N" "P" "T" "E" "A" "E" "L" "Q" "D" "M" "I" "N" "E" "V" "D" "A" "D" "G" "N" "G" "T" "I" "D" "F"

[67] "P" "E" "F" "L" "T" "M" "M" "E" "R" "K" "M" "K" "D" "T" "D" "S" "E" "E" "E" "I" "R" "E" "A" "F" "R" "V" "F" "D" "K" "D" "G" "N" "G"

[100] "Y" "I" "S" "A" "A" "K" "L" "R" "H" "V" "M" "T" "N" "L" "G" "E" "K" "L" "T" "D" "E" "E" "V" "D" "E" "M" "I" "R" "E" "A" "D" "A" "A"

[133] "G" "D" "G" "Q" "V" "N" "Y" "E" "E" "F" "V" "Q" "M" "M" "T" "A" "K"

> seq1 <- unlist(seq1)

> seq2 <- unlist(seq2)

> seq1 == seq2

[1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE

[23] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE FALSE TRUE TRUE TRUE TRUE TRUE TRUE TRUE

[45] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE

[67] TRUE TRUE TRUE TRUE TRUE TRUE TRUE FALSE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE

[89] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE FALSE TRUE TRUE TRUE TRUE TRUE

[111] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE FALSE FALSE

[133] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE

> z <- seq1 == seq2

> length(z[z==TRUE])

[1] 144

> length(z)

[1] 149

> percent\_identity <- 100\*length(z[z==TRUE])/length(z)

> percent\_identity

[1] 96.6443

**Aufgabe 2.4**

* Wenn gap penalty = -1 =>  
  2 Alignments:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| G | C | - | A | T | G |
| | |  |  |  | | |  |
| G | - | A | T | T | A |

Score = -2

und

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| G | C | A | - | T | G |
|  |  | | |  | | |  |
| - | G | A | T | T | A |

Score = -1

Bestes Aignment, wenn gap penalty = -1, ist das zweite Alignment, da der score größer ist.

* Wenn gap penalty = 0 =>

3 Alignments:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| G | C | A | T | G | - | - |
| | |  | | | | |  |  |  |
| G | - | A | T | - | T | A |

Score = 3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| G | C | A | - | T | - | G |
| | |  | | |  | | |  |  |
| G | - | A | T | T | A | - |

Score = 3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| G | C | A | T | - | - | G |
| | |  | | | | |  |  |  |
| G | - | A | T | T | A | - |

Score = 3

Alle drei Alignments mit gap penalty = 0 haben jeweils score=3, also sind gleich optimal.

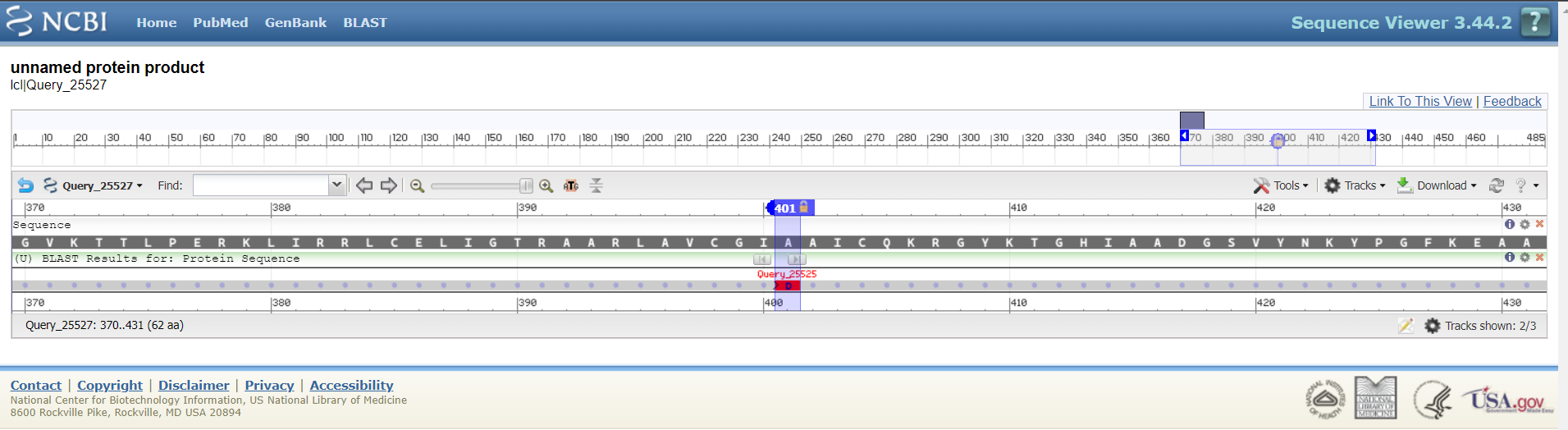
-> Vergleich zwischen den Alignments mit unterschiedlichen gap penalties: Wird die Lücke weniger bestraft, tendiert die Alignment länger zu werden.

**Aufgabe 2.5**

**2.5.1**

* Ich benutze die Variante blastp, da die Query und die Database Sequenzen Proteine sind.
* Es handelt sich wahrscheinlich um den Organismus Saccharomyces cerevisiae.
* Die Sequenz gehört wahrscheinlich zum Enzym Hexokinase 1.  
   hexokinase 1 [Saccharomyces cerevisiae S288C]
* Die Percentage Identity beträgt 99,79%.
* Der E-Score beträgt in diesem Fall 0. Er gibt die Anzahl der Treffer an, die zufällig gefunden werden könnten, da sie ähnlichen Score wie die Suchsequenz haben. Ein E-Score von 0 bedeutet, dass sich der Such-Treffer nicht durch Zufall erklären lässt, sondern sollte als biologisch relevant gelten. (Ich finde, dass E-Score und p value in der Statistik vom Prinzip her ähnlich sind.)

**2.5.2**



Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Es gibt eine Punktmutation in der gegebenen Aminosäurensequenz an der Position 401. Anstatt D (Asparaginsäure) wurde da A (Alanin) gebaut.