1	2	3	4	$\sum$ (7)

Bioinformatics I
WS 15/16
Tutor: Alexander Seitz

Jonas Ditz
& Benjamin Schroeder

Assignment 5

(Handed in 16. November 2015)

Theoretical Assignment - *Comparison with at most l mismatches*

General we are now only looking at the worst case scenario, which means uniform distributed mistmatches in an alignment. Also we only look at $k >= 1$, as $k = 0$ is in our application not a useful result. Assume two sequences of length t with l mismatches.
Than both sequences contain l tuples of length $L = \lfloor \frac{t}{l+1} \rfloor$ and one tuple which has a length of maximal $\lfloor \frac{t}{l+1} \rfloor$.

So both sequences share l or $l + 1$ k-tuples, if $k = \lfloor \frac{t}{l+1} \rfloor$, and for each $k \leq \lfloor \frac{t}{l+1} \rfloor$ they share $(l + 1) * \lfloor \frac{L}{k} \rfloor$ k-tuples.

Theoretical Assignment - *Linear programming by hand*

The feasible region of this linear program is shown in figure 1, where the red line is constraint 1, green line is constraint 2, constraint 3 is drawn as a blue line and the yellow line is constraint 4.

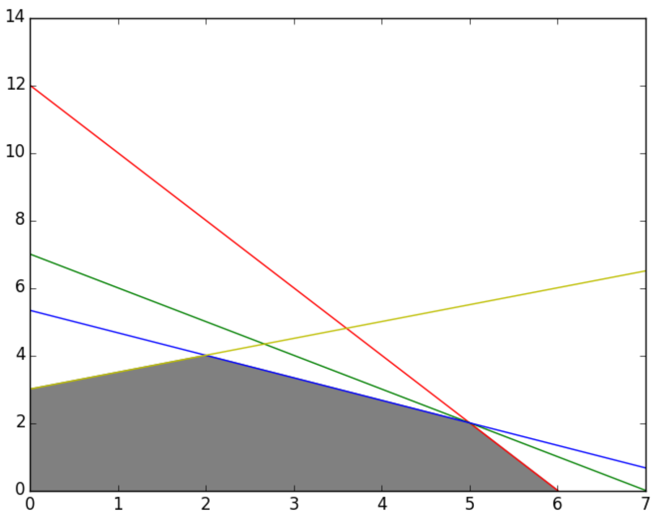


Figure 1: feasible region of linear program (grey area)

solution if t=s=1

Generally we only used integers in this task. The constraint for the algorithm are as following if t and s value is 1 now:

$$
\begin{align}
max.1*x_1 + 1*x_2 \tag{1}\\
s.t.2*x_1 + x_2 \leq 12 \tag{2}\\
2*x_1 + 3*x_2 \leq 7 \tag{3}\\
-1*x_1 + 2*x_2 \leq 16 \tag{4}\\
x_1, x_2 \geq 0 \tag{5}
\end{align}
$$

the next step we started to compute each line of the tuple. Starting with constraint 3 under the aspect of constraint 5. All possible tuples respectively to both constraints from the assignment are:
$temp1 = \{(0,7);(1,6);(2,5);(3,4);(4,3);(5,2);(6,1);(7,0)\}$

the constraint 2 was applied to temp1:
$temp2 = \{(0,7);(1,6);(2,5);(3,4);(4,3);(5,2);\}$

the next constraint (constraint 4) was applied:
$temp3 = \{(5,2)\}$

but not least the last constraint (4) is checked and the remaining tuple from the last step is valid:
$r = \{(5,2)\}$

the final result for the Algorithm with s=t=1 is the tuple: (5,2). Solving the formula with this numbers leads us to the following maximal value: $1*x_1 + 1*x_2 = 5 + 2 = 7$

How to make unsolvable?

To get a not solvable algorithm for example the following constraint could be added:

$$x_1 + x_2 \geq 7 \tag{6}$$

a new constraint just needs to be a contradiction to another constraint.

How to get infinite solutions with t and s?

The linear program has infinite solutions, if we assign the following values to s and t:

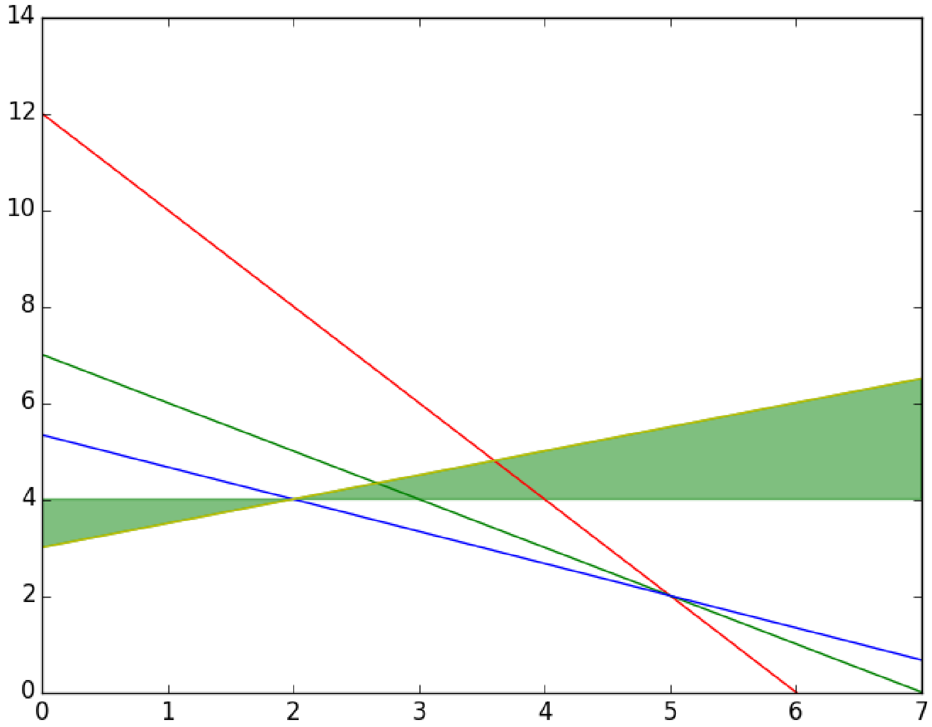
$$
\begin{gathered}
s = x_2\\
t = -x_1
\end{gathered} \tag{7}
$$

These values would lead to the equation $x_2 * x_1 + (-x_1) * x_2$, which is always zero. So the maximal result is zero and you can fill in arbitrary numbers to get this result, i.e. infinite solutions.

how to make the points a optimal solution?

For the first point $P_1 = (0, 3)$ we have to set $s = -1$ and $t = 1$. That result in a maximal value for (0,3) because we cannot set x_2 arbitrary big due to constraint 4. And if we assign a bigger value to x_1, would decrease the result. Also we would violate constraint 2, if x_1 is to big.

P2=(2,4)



To achive that P(2,4) is the only maximal value, we need to use the borders used in figure . The lower one is described by the maximal constraint (4) (equation 9)and the higher one is described by the horizontal line at $x2 = 4$. Next we can define relations, for the variables m_2 and c_2 of the searched linear equation. These are shown in equation 12 and .

$$
\begin{aligned}
-1x_1 + 2x_2 &= 6 && (8)\\
\Leftrightarrow x_2 &= \frac{1}{2}x_1 + 3 && (9)\\
= x_2 &= m_1x_1 + c_1 && (10)\\
& && (11)\\
m_1 > m_2 &> 0 && (12)\\
c_1 < c_2 &< 4 && (13)\\
& && (14)
\end{aligned}
$$

So any linear equation, which lies in the green area of figure , is a possible solution. We will now just choose one and define $c_2 = 3.5 = \frac{7}{2}$ according to relation . From this now we can deduce $m_2 = \frac{1}{4}$ as shown in Equation 16 so that P(2,4) is the only intersection with the feasible region.

$$
\begin{aligned}
4 &= m_2 * 2 + 3.5 && (15)\\
m_2 &= \frac{4 - 3.5}{2} = \frac{1}{4} && (16)\\
& && (17)
\end{aligned}
$$

The last step is now to get s and t, this is achieved by forming the found linear equation back to the desired form (Equation 20). The complete result is for $s = \frac{-1}{14}$ and for $t = \frac{2}{7}$.

$$
\begin{aligned}
x_2 &= \frac{1}{4} * x_1 + \frac{7}{2} && (18)\\
\Leftrightarrow -\frac{1}{4}x_1 + x_2 &= \frac{7}{2} && (19)\\
\Leftrightarrow -\frac{1}{14}x_1 + \frac{2}{7}x_2 &= 1 && (20)\\
\Rightarrow s = -\frac{1}{14}; t &= \frac{2}{7} && (21)\\
& && (22)
\end{aligned}
$$

P3=(1,3): We feel confident that there is no solution for this point.

To make $P_4 = (6, 0)$ the maximum we set $s = 1$ and $t = -1$. We cannot set x_1 to a bigger value than 6 without violating constraint 1.

Theoretical Assignment - *Bonus: Carillo-Lipman bound*

The Carrillo-Lipman bound is defined by $B_{pq} := B(A) - (\sum_{i<j} s(\hat{A}_i, \hat{A}_j) - s(\hat{A}_p, \hat{A}_q))$, where $B(A)$ is a score for the alignment computed by a heuristic and $s(\hat{A}_i, \hat{A}_j)$ is the score of a optimal alignment of sequences i and j. If we have just two sequences p and q, this equation would be simplified as

$$B_{pq} = B(A) - (s(\hat{A}_p, \hat{A}_q) - s(\hat{A}_p, \hat{A}_q)) = B(A).$$

So the Carrillo-Lipman bound for two sequences is just the computed score by a heuristic.