Weighted Interval Scheduling

\[
\begin{align*}
s_1 & \quad v_1 = 2 & f_1 \\
 s_2 & \quad v_2 = 4 & f_2 \\
 s_3 & \quad v_3 = 4 & f_3 \\
 s_4 & \quad v_4 = 7 & f_4 \\
 s_5 & \quad v_5 = 2 & f_5 \\
 s_6 & \quad v_6 = 1 & f_6
\end{align*}
\]

Either \( s_6 \) \( v_6 \) \( f_6 \) is selected, or it is not.
There is a set of intervals not overlapping with \( s_6 \quad v_6 \quad f_6 \). Call the set \( P(6) \).

If \( s_6 \quad v_6 \quad f_6 \) is selected, we get \( v_6 \) and must do optimally on \( P(6) \). Write this as \( v_6 + M[P(6)] \).

If not selected, \( M[5] \).

\( M[n] \) is the optimal VALUE.
Weighted Interval Scheduling

- \( M[0] = 0 \)
- \( M[1] = v_1 \)
- \( M[2] = \max \left\{ v_2 + M[P(2)], M[1] \right\} \)
- \( M[3] = \max \left\{ v_3 + M[P(3)], M[2] \right\} \)

- Use the \textit{VALUE} of sub-solutions (NOT the sub-solution itself) to get larger sub-solution values.

- E.g. \( F(n) = \max \left\{ a_n + F(n - 2), F(n - 1) \right\} \)
Weighted Interval Scheduling

In General:

- $M[n] = \max \left\{ v_n + M[P(n)] \right.$
  \left. \begin{array}{c}
  M[n - 1]
  \end{array} \right\}$

- $M[j] = \max \left\{ v_j + M[P(j)] \right.$
  \left. \begin{array}{c}
  M[j - 1]
  \end{array} \right\}$
- An order or subproblems
- Sub-solution values not structure of subproblems
Given a sequence of points \( p_i = (x_i, y_i) \), find \( a \) and \( b \) so that the sum of the error\(^2\) is minimized in \( y = b + ax \).

\[
\begin{align*}
  p_0 &= (0, 2.1) \\
  p_1 &= (1, 2.9) \\
  p_2 &= (2, 4.1) \\
  p_3 &= (3, 4.9)
\end{align*}
\]
Use two line-segments

\[
\text{error} = \sum_{i=1}^{n} (y_i - ax_i - b)^2
\]

Want \( opt(1, n) \)
Least Squares

- If we use \((n - 1)\) line segments, we have no error.
- If we change directions of line segments, each change costs, say, $3

\[
\text{opt}(1, n) = \text{opt}(1, i - 1) + c + e_{i,n}
\]