Assignment 01

DSA2101 AY 22/23 Sem I

Dataset

The data you need for this assignment is in the `assignment01.rds` file in your folder on Canvas.

Akima interpolation

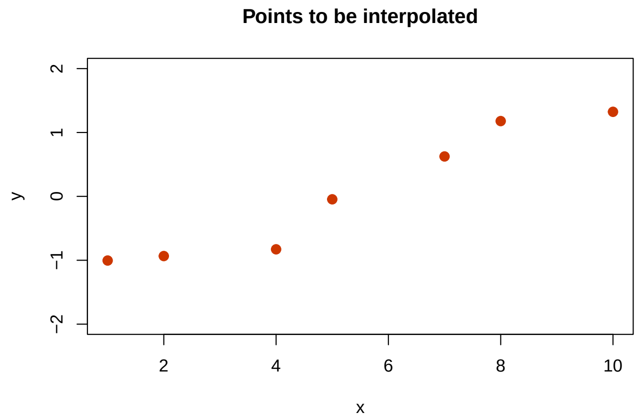
Akima interpolation is a technique for interpolating between two points. It is widely used, even today. The interpolation is done by fitting a cubic polynomial between the two points.

There are two steps involved in estimating the polynomial between two points:

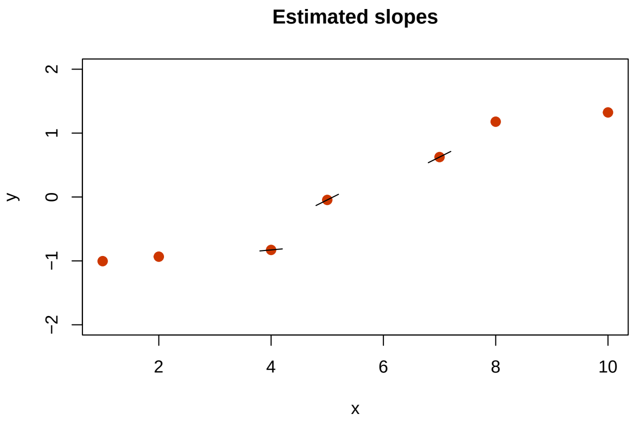
1. The slopes at each of the two points have to be estimated. At each point, this is carried out using the nearest two points on each side.
2. The slopes are then used to fit the cubic polynomial between (x_1, y_1) and (x_2, y_2).

Example

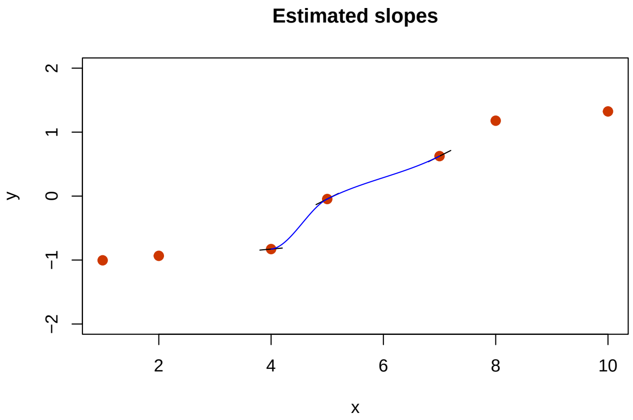
Suppose we wish to interpolate these seven points:



Here is what the estimated slopes for the middle points look like. The algorithm can be modified to interpolate the extreme two points as well, but we will leave it out for this assignment.



Finally, we use the estimated slopes and the observed coordinates to fit a separate cubic polynomial between each pair of adjacent points.



Question 1

Write a function `compute_slopes` that takes in two vectors of length 5, `x` and `y`, and returns the estimated slope at `x[3]`. Here is the algorithm for it:

1. We are given $(x_1, y_1), (x_2, y_2), \ldots, (x_5, y_5)$ where $x_1 < x_2 < \ldots < x_5$. Our goal is to estimate the slope of the interpolated function at x_3.
2. Compute the gradients of the 4 linear interpolations between the points. For instance,

$$m_1 = \frac{y_2 - y_1}{x_2 - x_1},\ m_2 = \frac{y_3 - y_2}{x_3 - x_2},\ \text{and so on}$$

3. The slope at x_3 is computed to be:

$$s = \frac{|m_4 - m_3|m_2 + |m_2 - m_1|m_3}{|m_4 - m_3| + |m_2 - m_1|}$$

4. In the rare situation that $m_1 = m_2$ **and** $m_3 = m_4$, we set

$$s = \frac{1}{2}(m_2 + m_3)$$

Here is a check of how it should work:

```
x_eg <- -2:2
y_eg <- x_eg^2
compute_slopes(x_eg,y_eg)
```

```
## [1] 0
```

Question 2

Write a function `est_poly` that will return a set of interpolated coordinates between two points (x_1, y_1) and (x_2, y_2). The arguments to this function should be:

1. `x`: a vector containing x_1 and x_2
2. `y`: a vector containing y_1 and y_2
3. `slopes`: A vector containing estimated slopes at (x_1, y_1) and (x_2, y_2) (using `compute_slopes()`).
4. `n`: The number of interpolated points to return. The return value should be a matrix with two columns.

The task is: given (x_1, y_1) and (x_2, y_2), and that the estimated gradients at the two points are s_1 and s_2, we wish to estimate a cubic polynomial that passes through the two points, and has derivatives s_1 and s_2 at the two points.

Here is the algorithm to implement:

1. The form of the polynomial will be

$$y = p_0 + p_1(x - x_1) + p_2(x - x_1)^2 + p_3(x - x_1)^3$$

2. The solutions for the coefficients are:
   - $p_0 = y_1$
   - $p_1 = s_1$
   - $p_2 = [3(y_2 - y_1)/(x_2 - x_1) - 2s_1 - s_2]/(x_2 - x_1)$
   - $p_3 = [s_1 + s_2 - 2(y_2 - y_1)/(x_2 - x_1)]/(x_2 - x_1)^2$

Here is an example of how it should work:

```
x_eg2 <- c(0, 1)
y_eg2 <- c(0, 1)
slopes_eg <- c(0, 2)
est_poly(x_eg2, y_eg2, slopes_eg, n=5)
```

```
##      x_vals
## [1,]   0.00 0.0000
## [2,]   0.25 0.0625
## [3,]   0.50 0.2500
## [4,]   0.75 0.5625
## [5,]   1.00 1.0000
```

Question 3

Use the two functions you have written to interpolate and plot the points in `akima_data`.

Cleaning up Artist Data

The `artist_data` contains information on artists whose work was exhibited at a museum in England. Use functions in the `stringr` package to:

Questions 4, 5, and 6

4. Clean up the `gender` and `dates` columns as best you can. Both contain misspellings and ambiguities. Store your cleaned data frame in a data frame called `artist_data_clean`
5. Extract the unique set of unambiguous birth years for the artists. Store the *integer* in a vector `artist_birth_years`.

6. Extract the unique set of country of birth for all artists. There is no need to change the names to modern English. Store the extracted names in a character vector named `artist_birth_country`

Requirements:

1. Your Rmd file must knit on our machines. It should create a html with a properly labelled plot.
2. Avoid "for" loops as much as possible.
3. Use vectorised operations in R.
4. If we run all the chunks in your Rmd file, we should obtain:
    - functions named `est_poly()` and `compute_slopes`
    - a data frame named `artist_data_clean`,
    - vectors named `artist_birth_years` and `artist_birth_country`.

References

1. A new method of interpolation and smooth curve fitting based on local procedures (1970), H Akima, *J. of the Association for Computing Machinery.*