

Оглавление

1	Интерфейс библиотеки	2
1.1	Функция <code>interpolate</code>	2
1.2	Функция <code>interpolate-parametric</code>	3
1.3	Функция <code>interpolate-grid</code>	4

1 Интерфейс библиотеки

Разработанная библиотека предоставляет 3 функции:

`(interpolate points type & options)`

`(interpolate-parametric points type & options)`

`(interpolate-grid grid type & options)`

Все вышеперечисленные функции имеют сходную сигнатуру. Они принимают следующие параметры:

- **points** или **grid** - набор точек или сетка, которые требуется интерполировать;
- **type** - тип интерполяции;
- **options** - дополнительные опции, специфичные для каждой функции и типа интерполяции;

Все функции возвращают новую функцию - интерполирующую функцию, с помощью которой находятся значения в интересующих точках.

1.1 Функция `interpolate`

Данная функция позволяет построить интерполирующую функцию $f(x) = y$ по заданному набору точек (x_i, y_i) . Точки могут задаваться в любом порядке, перед использованием они будут автоматически отсортированы по координате x .

Данная функция поддерживает следующие типы интерполяции: линейная, полиномиальная, кубический сплайн, кубический Эрмитов сплайн, среднеквадратичное приближение. Соответствующие аргументы для параметра `type`: `:linear`, `:polynomial`, `:cubic`, `:cubic-hermite`, `:linear-least-squares`.

Дополнительные опции:

- **:boundaries** - граничные условия для кубического сплайна. Поддерживаются 2 вида условий: естественные (`:natural`) и замкнутые (`:closed`);
- **:derivatives** - производные для кубического Эрмитова сплайна;
- **:basis**, **:n**, **:degree** - опции настройки среднеквадратичного приближения. Позволяют задать базис, произвольный или один из 2 встроенных (полиномиальный и В-сплайны); число функций в базисе, если выбран встроенный; степень В-сплайнов;

Пример:

Построение кубического сплайна с замкнутыми граничными условиями по точкам $(0, 0)$, $(1, 3)$, $(2, 0)$, $(4, 2)$:

```
(def points [[0 0] [1 3] [2 0] [4 2]])

(def cubic (interpolate points :cubic :boundaries :closed))

(cubic 0) ; 0.0
(cubic 1) ; 3.0
(cubic 3) ; -1.2380952380952381
```

1.2 Функция `interpolate-parametric`

Данная функция строит интерполирующую параметрическую функцию $f(t) = (x^1, x^2, \dots, x^n)$ определённую на отрезке $[0, 1]$ и проходящую через заданные пользователем точки $(x_i^1, x_i^2, \dots, x_i^n)$. Особенность данной функции заключается в том, что пользователь не задаёт узлы t_i , соответствующие каждой точке, данные узлы они выбираются алгоритмом. Эта особенность используется в полиномиальной интерполяции, для которой очень важен выбор хороших узлов: для параметрической полиномиальной интерполяции используются узлы Чебышева.

Данная функция поддерживает такие же типы интерполяции, как `interpolate`, а также В-сплайн (`:b-spline`).

Дополнительные опции:

Поддерживается тот же набор дополнительных опций, которые поддерживает функция `interpolate`, а также добавлены 2 новые опции:

- `:degree` - задаёт степень В-сплайна, если выбрана аппроксимация В-сплайнами;
- `:range` - отрезок, на котором будет определена интерполирующая функция, по умолчанию это $[0, 1]$;

Пример:

Построение параметрической интерполирующей функции на отрезке $[-1, 1]$ по точкам $(0, 0)$, $(1, 3)$, $(2, 0)$, $(4, 2)$:

```
(def points [[0 0] [1 3] [2 0] [4 2]])
```

```
(def linear (interpolate-parametric points :linear
:range [-1 1]))

(linear -1) ; (0.0 0.0)
(linear 1) ; (4.0 2.0)
(linear 0) ; (1.5 1.5)
```

1.3 Функция interpolate-grid

Данная функция строит интерполирующую функцию 2 переменных $f(x, y) = z$ по прямоугольной сетке точек. По умолчанию, пользователь задаёт сетку значений $\{z_{ij}\}, i = \overline{1, n}, j = \overline{1, m}$, далее автоматически строится равномерная сетка узлов размера $n \times m$ заданная на области $[0, 1] \times [0, 1]$. В конце полученная сетка узлов и значений интерполируется и строится интерполирующая функция.

Функция поддерживает следующие типы интерполяций: билинейная (:bilinear), полиномиальная (:polynomial), бикубический сплайн (:bicubic), бикубический Эрмитов сплайн (:bicubic-hermite), среднеквадратичное приближение (:linear-least-squares), В-сплайновая поверхность (:b-surface).

Дополнительные опции:

- **:boundaries** - граничные условия для бикубического сплайна. Поддерживаются 2 вида условий: естественные (:natural) и замкнутые (:closed);
- **:basis, :n** - опции настройки среднеквадратичного приближения. Позволяют задать базис, произвольный или встроенный полиномиальный и число функций в базисе, если выбран встроенный;
- **:degree** - задаёт степень В-сплайна, если выбрана аппроксимация В-сплайновой поверхностью;
- **:x-range, :y-range** - область, на которой задана интерполирующая функция, по умолчанию это $[0, 1] \times [0, 1]$;
- **:xs, :ys** - явное задание узлов интерполяции;

Пример:

Интерполирование сетки значений $\begin{pmatrix} 0 & 1 & 2 \\ 3 & 4 & 5 \\ 6 & 7 & 8 \end{pmatrix}$ с использованием билинейной интерполяции:

```
(def grid [[0 1 2]
           [3 4 5]
           [6 7 8]])

(def bilinear (interpolate-grid grid :bilinear))

(bilinear 0 0) ; 0.0
(bilinear 1 1) ; 8.0
(bilinear 0.5 0.5) ; 4.0
(bilinear 0.25 1) ; 6.5
```