

Построение разряженной монотонной регрессии методом Франка-Вульфа

С.П.Сидоров, А.Р.Файзлиев, А.А.Гудков, М.А.Левшунов

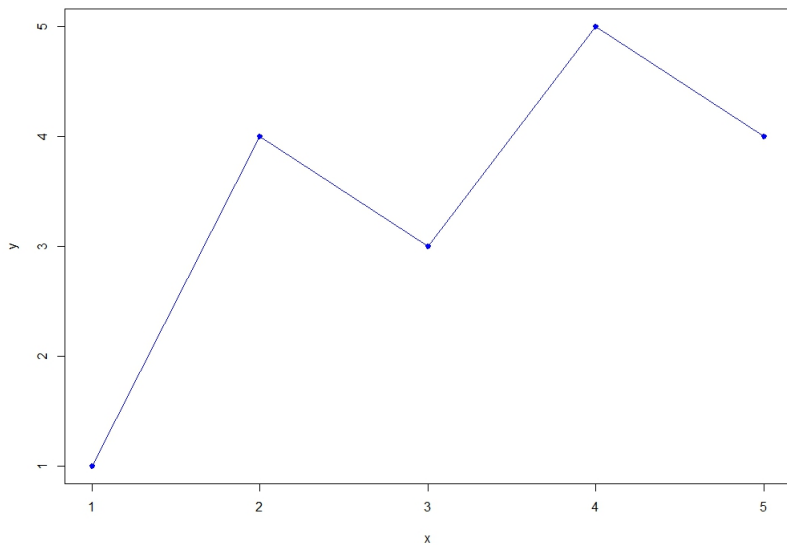
Саратовский национальный исследовательский
государственный университет
им. Н. Г. Чернышевского

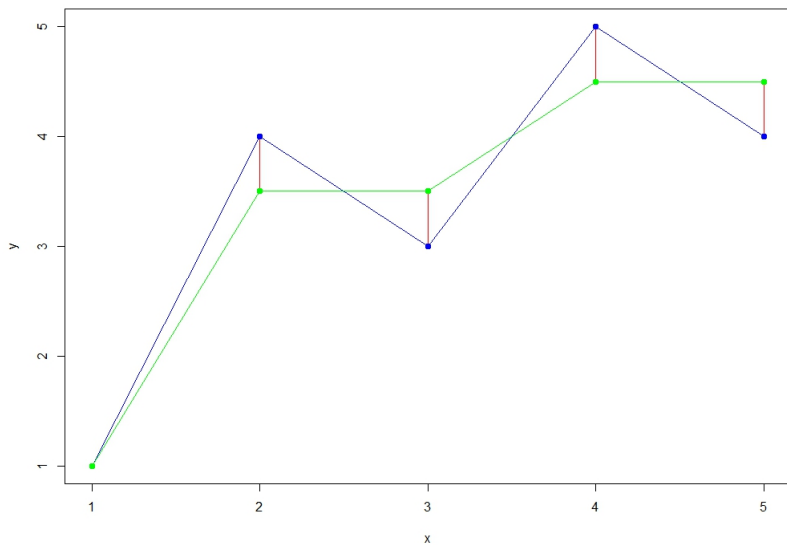
2017г.

Вектор $y \in \mathbb{R}^n$ задан, найти вектор $z \in \mathbb{R}^n$, удовлетворяющий:

$$f(z) = \frac{1}{n} \sum_{i=1}^n (z_i - y_i)^2 \rightarrow \min, \quad (1)$$

$$z : (z_{i+1} \geq z_i, i = 1, \dots, n-1).$$





Замена: $\zeta_1 = z_1, \zeta_{i+1} = z_{i+1} - z_i, i = 1, \dots, n-1$,
эквивалентная (1) задача:

$$g(\zeta) = \frac{1}{n} \sum_{i=1}^n \left(\sum_{j=1}^i \zeta_j - y_i \right)^2 \rightarrow \min_{\zeta \in S}. \quad (2)$$

$S = \{\zeta = (\zeta_1, \zeta_2, \dots, \zeta_n) \in \mathbb{R}^n \mid \zeta_1 \in \mathbb{R}, \zeta_i \geq 0, i = 2, \dots, n\}.$

Хорошо известно, что задача (2) является NP-сложной задачей (Leeuw, J., Hornik, K., P., M.: Isotone optimization in r: Pool-adjacent-violators algorithm (PAVA) and active set methods. Journal of Statistical Software 32(5), 1–24, 2009).

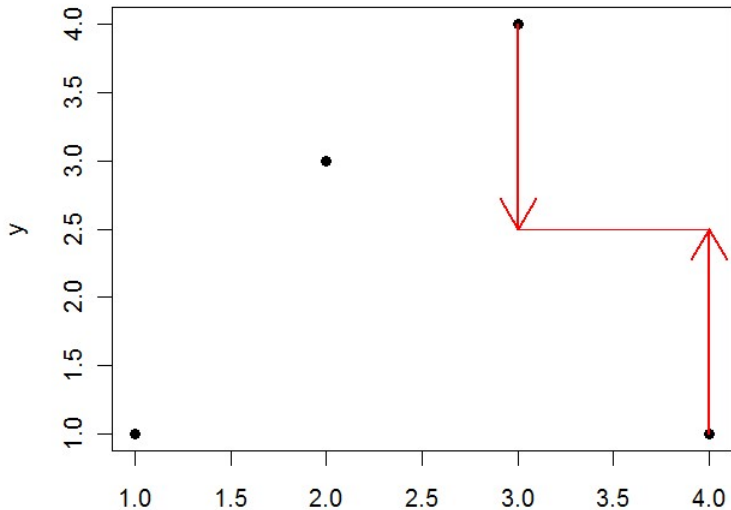
Основной вклад - Robertson и Dykstra (1988 год).
Barlow and Brunk (1972 год) и Dykstra (1981 год),
как и Best, Chakravarti, Ubhaya (2000)
рассматривали монотонную регрессию как задачу
квадратичного и выпуклого программирования.

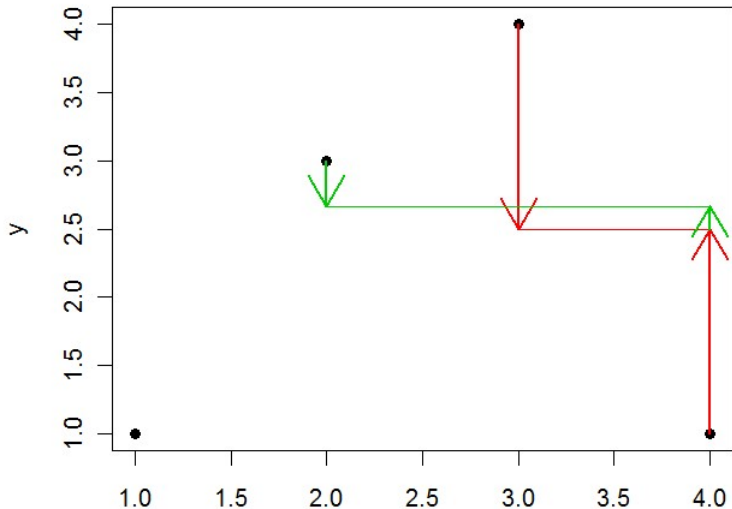
Монотонная регрессия применяется:

- 1 для восстановления функции распределения;
- 2 для получения средних экспериментальных результатов;
- 3 в неметрическом многомерном шкалировании;
- 4 при сглаживании эмпирических данных.

Наиболее известным алгоритмом для решения задачи (2) является Pool Adjacent-Violators Algorithm (PAVA) (Barlow, Bartholomew, Bremner, Brunk 1972).

- ❶ На итерации $i = 0$ все $z_j^0 := y_j$ (здесь и далее верхний индекс - номер итерации).
- ❷ Пронумеруем блоки $r = 1, \dots, B$, где на 0 итерации $B := n$, т. е. каждое наблюдение z_r^0 формирует блок.
- ❸ Объединим значения z^i в блок, если $z_{r+1}^i < z_r^i$.
- ❹ Решим $g(z)$ отдельно для каждого блока, получив значения z_r^{i+1} .
- ❺ Если найдется $z_{r+1}^{i+1} < z_r^{i+1}$, то $i := i + 1$ и вернемся на шаг 3.





\mathbb{D} - компактное выпуклое подмножество
векторного пространства \mathbb{R}^n .

f - выпуклая и непрерывно дифференцируемая в
 \mathbb{D} функция.

Основная задача:

$$\inf_{x \in \mathbb{D}} f(x). \quad (3)$$

В 1956 году Франк и Вульф разработали алгоритм решения задач квадратичного программирования с линейными ограничениями.

$x^{(0)}$ - начальная точка.

На k -ом шаге рассматриваем задачу минимизации:

$$Z_k(y) := f(x^{(k)}) + \nabla f(x^{(k)})^T (y - x^{(k)}), \quad (4)$$

при условии $y \in \mathbb{D}$.

Минимизация эквивалентна задаче:

$$\min_{y \in \mathbb{D}} \nabla f(x^{(k)})^T (y - x^{(k)}).$$

Алгоритм решения задачи (3):

- ❶ Задаём $x^{(0)} \in \mathbb{D}$, N - количество итераций;
- ❷ В цикле по $k = 0, 1, \dots, N$:
 - ❶ Вычисляем $y := \arg \min_{y \in \mathbb{D}} \nabla f(x^{(k)})^T y$;
 - ❷ Переходим к следующей точке

$$x^{(k+1)} = (1 - \alpha)x^{(k)} + \alpha y, \text{ где } \alpha := \frac{2}{k+2};$$

- ❸ Завершаем цикл.

Обозначим:

$$S := \left\{ \zeta \in \mathbb{R}^n : \zeta_1 \geq \min_{j=1,\dots,n} y_j, \zeta_i \geq 0, i = 2, \dots, n, \sum_{k=1}^n \zeta_k \leq \max_{j=1,\dots,n} y_j \right\}, \quad (5)$$

$\nabla g(\zeta) = \left(\frac{\partial g}{\partial \zeta_1}, \frac{\partial g}{\partial \zeta_2}, \dots, \frac{\partial g}{\partial \zeta_n} \right)$ – градиент функции g в точке ζ , где

$$\frac{\partial g}{\partial \zeta_k} = \frac{2}{n} \sum_{i=k}^n \left(\zeta_1 + \sum_{j=2}^i \zeta_j - y_i \right), k = 1, \dots, n. \quad (6)$$

- ❶ Зададим N – максимальное число шагов цикла.
- ❷ Положим $t = 0$, зададим начальную точку алгоритма

$$\zeta^0 = \left(\frac{1}{n} \sum_{i=1}^n y_i, 0, \dots, 0 \right).$$

- ❸ Пока $t < N$:

- ❶ Вычисляем $\nabla g(\zeta^t)$. Решаем задачу линейной оптимизации:

$$\nabla g(\zeta^t)^T \zeta \rightarrow \min, \zeta \in S. \quad (7)$$

- ❷ $\tilde{\zeta}^t$ - решение задачи (7).

- ❸ Положим $\zeta^{t+1} = (1 - \alpha)\zeta^t + \alpha\tilde{\zeta}^t, \alpha = \frac{2}{t+2}$.

- ❹ $t = t + 1$, переходим к следующему шагу.

- ❹ Находим искомую 1-монотонную последовательность $z = (z_1, \dots, z_n)$ из вектора ζ^N .

Нами была доказана следующая теорема:

Theorem

Пусть значения $\{\zeta^t\}$ получены с помощью жадного алгоритма типа Франка-Вульфа, тогда для $t \geq 1$ справедлива оценка:

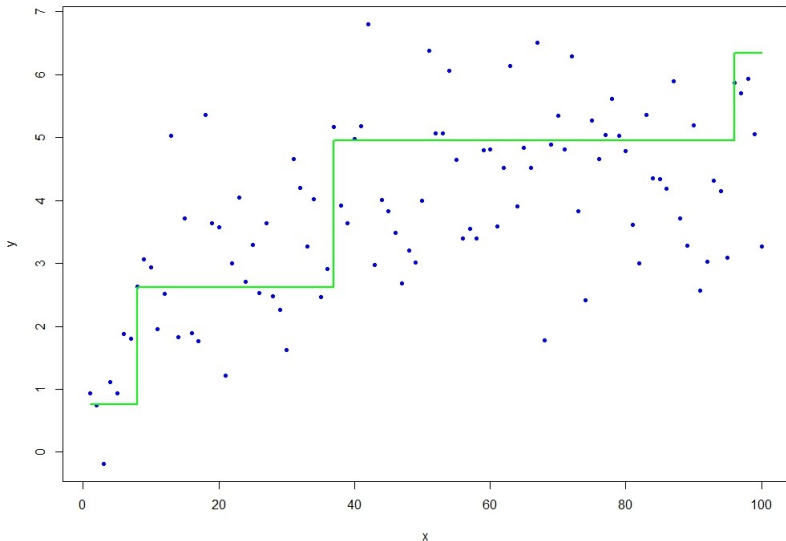
$$g(\zeta^t) - g^* \leq \frac{2M(\text{Diam}(S))^2}{t+2}.$$

g^* - точное решение задачи,

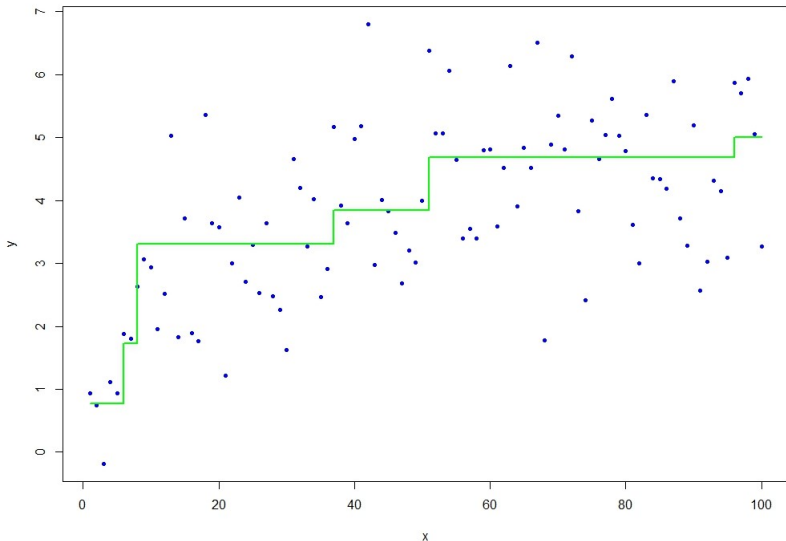
$$\text{Diam}(S) = \sqrt{2}(\max_i y_i - \min_i y_i),$$

$$M = 2\sqrt{\frac{(n+1)(2n+1)}{6n}}.$$

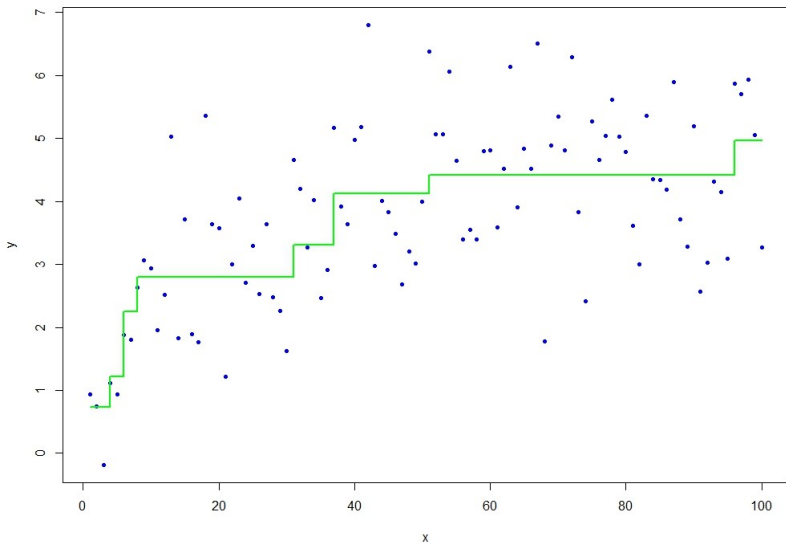
Результат работы Жадного алгоритма для 5 итераций.



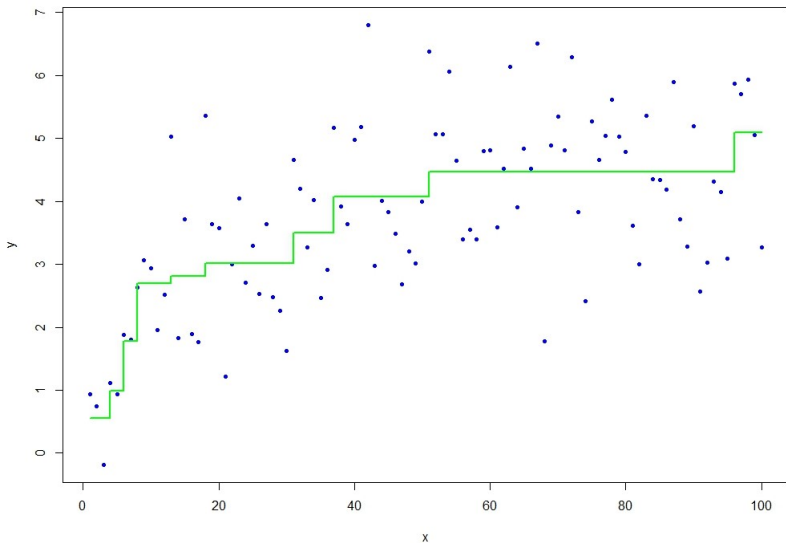
Результат работы Жадного алгоритма для 10 итераций.



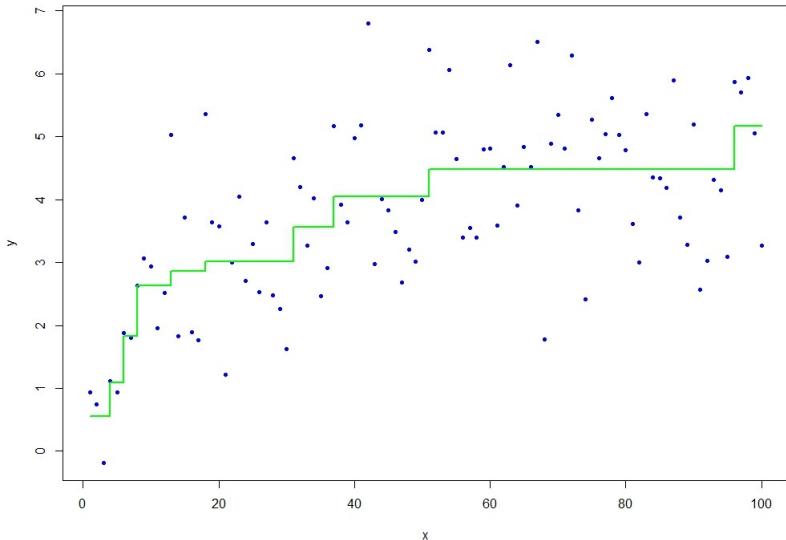
Результат работы Жадного алгоритма для 20 итераций.



Результат работы Жадного алгоритма для 50 итераций.



Результат работы Жадного алгоритма для 100 итераций.



Набор точек имеет следующую структуру:

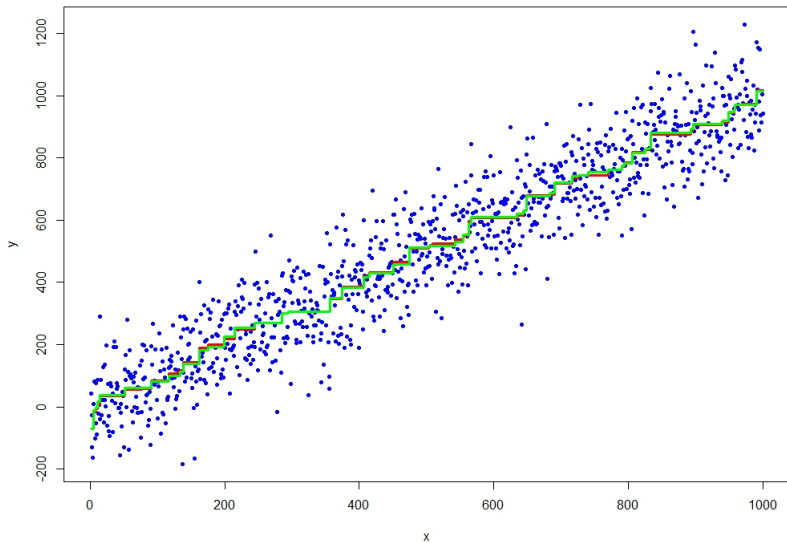
$x = 1 : 1000, y = x + N(0, 100)$.

Относительная ошибка $= \frac{f - f^*}{f^*}$,

f - значение функции $f(z) = \frac{1}{n} \sum_{i=1}^n (z_i - y_i)^2$ для вектора

$z = (z_1, \dots, z_n)$, полученного с помощью алгоритма.

f^* - значение функции $f(z)$ для вектора z , который является точным решением задачи (1).



Алгоритм (число итераций)	Относитель- ная ошибка	Кардиналь- ность	Затраченное время
PAVA	0	54	0.002
Greedy(50)	0.0241	27	0.072
Greedy(100)	0.0075	40	0.105
Greedy(200)	0.0024	47	0.169
Greedy(500)	0.0003	53	0.379
Greedy(1000)	0.00007	53	0.784
Greedy(2000)	0.00002	54	1.433

Набор точек имеет следующую структуру:

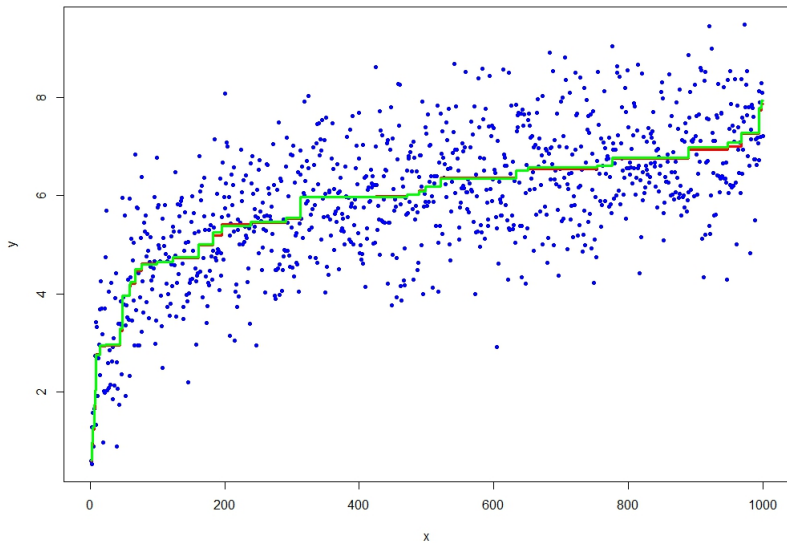
$$x = 1 : 1000, y = \ln(x) + N(0, 1).$$

$$\text{Относительная ошибка} = \frac{f - f^*}{f^*},$$

$$f - \text{значение функции } f(z) = \frac{1}{n} \sum_{i=1}^n (z_i - y_i)^2 \text{ для}$$

вектора $z = (z_1, \dots, z_n)$, полученного с помощью алгоритма.

f^* - значение функции $f(z)$ для вектора z , который является точным решением задачи (1).



Алгоритм (число итераций)	Относитель- ная ошибка	Кардиналь- ность	Затраченное время
PAVA	0	31	0.001
Greedy(50)	0.0136	18	0.061
Greedy(100)	0.0032	25	0.091
Greedy(200)	0.0009	26	0.149
Greedy(500)	0.0001	30	0.333
Greedy(1000)	0.0001	31	0.663
Greedy(2000)	0	31	1.259

Набор точек имеет следующую структуру:

$$x = 1 : 1000,$$

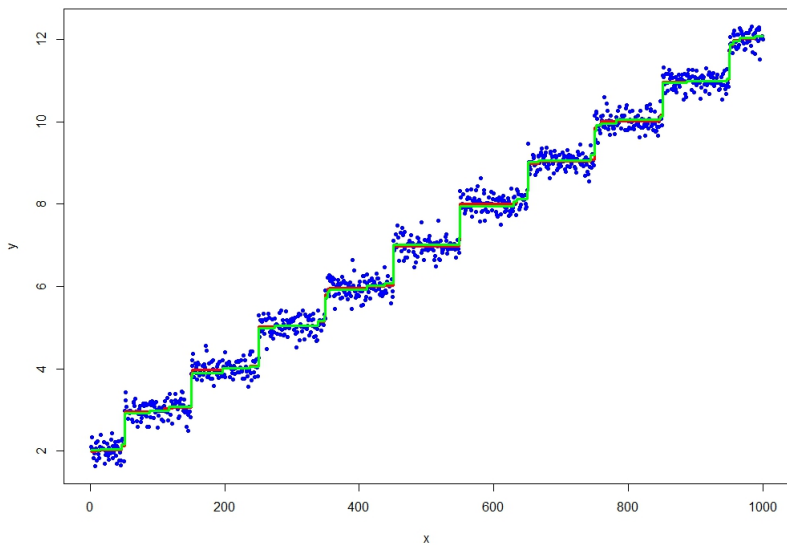
y - ступенчатая функция с отклонением $N(0, 0.2)$.

$$\text{Относительная ошибка} = \frac{f - f^*}{f^*},$$

f - значение функции $f(z) = \frac{1}{n} \sum_{i=1}^n (z_i - y_i)^2$ для

вектора $z = (z_1, \dots, z_n)$, полученного с помощью алгоритма.

f^* - значение функции $f(z)$ для вектора z , который является точным решением задачи (1).



Алгоритм (число итераций)	Относитель- ная ошибка	Кардиналь- ность	Затраченное время
PAVA	0	56	0.003
Greedy(50)	0.4242	20	0.059
Greedy(100)	0.1025	33	0.106
Greedy(200)	0.0317	44	0.151
Greedy(500)	0.0039	49	0.339
Greedy(1000)	0.00105	52	0.640
Greedy(2000)	0.00026	54	1.277

- ❶ Разработать алгоритм построения разряженной монотонной регрессии типа Франка-Вульфа для многомерного случая.
- ❷ Разработать алгоритм типа Франка-Вульфа для построения кусочно-линейной функции.
- ❸ Приложение к задачам формосохраняющего динамического программирования.

СПАСИБО ЗА ВНИМАНИЕ!