

Липецкий государственный технический университет

Кафедра прикладной математики

## **КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ МАТЕМАТИЧЕСКИХ ИССЛЕДОВАНИЙ**

Лекция 11

### **Алгоритмы кластерного анализа**

Составитель - Сысоев А.С., к.т.н., доцент

Липецк - 2021

## Outline

---

- 11.1. Задачи кластерного анализа
- 11.2. Эвристические графовые алгоритмы
- 11.3. Функционалы качества кластеризации
- 11.4. Статистические алгоритмы
- 11.5. Иерархическая кластеризация
- 11.6. Определение числа кластеров
- 11.7. Кластеризация в R

## 11.1. Задачи кластерного анализа

---

Задача кластеризации (обучения без учителя) заключается в следующем. Имеется обучающая выборка  $X^e = \{x_1, \dots, x_\ell\} \subset X$  и функция расстояния между объектами  $\rho(x, x')$ . Требуется разбить выборку на непересекающиеся подмножества, называемые *кластерами*, так, чтобы каждый кластер состоял из объектов, близких по метрике  $\rho$ , а объекты разных кластеров существенно отличались. При этом каждому объекту  $x_i \in X^e$  приписывается метка (номер) кластера  $y_i$ .

Алгоритм кластеризации — это функция  $a: X \rightarrow Y$ , которая любому объекту  $x \in X$  ставит в соответствие метку кластера  $y \in Y$ . Множество меток  $Y$  в некоторых случаях известно заранее, однако чаще ставится задача определить оптимальное число кластеров, с точки зрения того или иного критерия качества кластеризации.

Решение задачи кластеризации принципиально неоднозначно:

- не существует однозначно наилучшего критерия качества кластеризации,
- число кластеров неизвестно заранее и устанавливается в соответствии с некоторым субъективным критерием,
- результат кластеризации существенно зависит от метрики  $\rho$ .

## 11.1. Задачи кластерного анализа

---

Цели кластеризации могут быть различными в зависимости от особенностей конкретной прикладной задачи:

- Понять структуру множества объектов  $X^e$ , разбив его на группы схожих объектов. Упростить дальнейшую обработку данных и принятия решений, работая с каждым кластером по отдельности (стратегия «разделяй и властвуй»).
- Сократить объём хранимых данных в случае сверхбольшой выборки  $X^e$ , оставив по одному наиболее типичному представителю от каждого кластера.
- Выделить нетипичные объекты, которые не подходят ни к одному из кластеров. Эту задачу называют одноклассовой классификацией, обнаружением нетипичности или новизны (novelty detection).

## 11.2. Эвристические графовые алгоритмы

---

Вершинам графа соответствуют объекты выборки, а рёбрам — попарные расстояния между объектами  $p_{ij} = p(x_i, x_j)$ .

**Алгоритм выделения связных компонент.** Задаётся параметр  $R$  и в графе удаляются все рёбра  $(i, j)$ , для которых  $p_{ij} > R$ . Соединёнными остаются только наиболее близкие пары объектов. Идея алгоритма заключается в том, чтобы подобрать такое значение  $R \in [\min p_{ij}, \max p_{ij}]$ , при котором граф развалится на несколько связных компонент. Найденные связные компоненты — и есть кластеры.

*Связной компонентой графа* называется подмножество его вершин, в котором любые две вершины можно соединить путём, целиком лежащим в этом подмножестве. Для поиска связных компонент можно использовать стандартные алгоритмы поиска в ширину (алгоритм Дейкстры) или поиска в глубину.

Недостатки:

- ограниченная применимость,
- плохая управляемость числом кластеров.

## 11.2. Эвристические графовые алгоритмы

---

**Алгоритм кратчайшего незамкнутого пути** строит граф из  $\ell-1$  рёбер так, чтобы они соединяли все  $\ell$  точек и обладали минимальной суммарной длиной. Такой граф называется *кратчайшим незамкнутым путём*, минимальным покрывающим деревом или каркасом.

---

### Алгоритм 1.1. Алгоритм кратчайшего незамкнутого пути (КНП)

---

- 1: Найти пару точек  $(i, j)$  с наименьшим  $\rho_{ij}$  и соединить их ребром;
  - 2: **пока** в выборке остаются изолированные точки
  - 3:   найти изолированную точку, ближайшую к некоторой неизолзированной;
  - 4:   соединить эти две точки ребром;
  - 5: удалить  $K - 1$  самых длинных рёбер;
- 

Недостатки:

- ограниченная применимость,
- высокая трудоёмкость.

## 11.2. Эвристические графовые алгоритмы

---

### Алгоритм FOREL (ФОРмальный Элемент)

---

#### Алгоритм 1.2. Алгоритм FOREL

---

- 1: Инициализировать множество некластеризованных точек:  
 $U := X^\ell$ ;
  - 2: **пока** в выборке есть некластеризованные точки,  $U \neq \emptyset$ :
  - 3:   взять произвольную точку  $x_0 \in U$  случайным образом;
  - 4:   **повторять**
  - 5:     образовать кластер — сферу с центром в  $x_0$  и радиусом  $R$ :  
       $K_0 := \{x_i \in U \mid \rho(x_i, x_0) \leq R\}$ ;
  - 6:     поместить центр сферы в центр масс кластера:  
       $x_0 := \frac{1}{|K_0|} \sum_{x_i \in K_0} x_i$ ;
  - 7:     **пока** центр  $x_0$  не стабилизируется;
  - 8:     пометить все точки  $K_0$  как кластеризованные:  
       $U := U \setminus K_0$ ;
  - 9:   применить алгоритм КНП к множеству центров всех найденных кластеров;
  - 10: каждый объект  $x_i \in X^\ell$  приписать кластеру с ближайшим центром;
- 

Недостатки:

- чувствителен к выбору начального положения точки  $x_0$  для каждого нового кластера.

### 11.3. Функционалы качества кластеризации

---

Задачу кластеризации можно ставить как задачу дискретной оптимизации: необходимо так приписать номера кластеров  $y_i$  объектам  $x_i$ , чтобы значение выбранного функционала качества приняло наилучшее значение.

- Среднее внутрикластерное расстояние должно быть как можно меньше:

$$F_0 = \frac{\sum_{i < j} [y_i = y_j] \rho(x_i, x_j)}{\sum_{i < j} [y_i = y_j]} \rightarrow \min .$$

- Среднее межкластерное расстояние должно быть как можно больше:

$$F_1 = \frac{\sum_{i < j} [y_i \neq y_j] \rho(x_i, x_j)}{\sum_{i < j} [y_i \neq y_j]} \rightarrow \max .$$

- Сумма средних внутрикластерных расстояний должна быть как можно меньше:

$$\Phi_0 = \sum_{y \in Y} \frac{1}{|K_y|} \sum_{i: y_i = y} \rho^2(x_i, \mu_y) \rightarrow \min$$

- Сумма межкластерных расстояний должна быть как можно больше:

$$\Phi_1 = \sum_{y \in Y} \rho^2(\mu_y, \mu) \rightarrow \max$$



## 11.4. Статистические алгоритмы

---

Статистические алгоритмы основаны на предположении, что кластеры неплохо описываются некоторым семейством вероятностных распределений. Тогда задача кластеризации сводится к разделению смеси распределений по конечной выборке.

### ЕМ-алгоритм

*Гипотеза о вероятностной природе данных.* Объекты выборки  $\mathbf{X}^{\ell}$  появляются случайно и независимо согласно вероятностному распределению, представляющему собой смесь распределений

$$p(x) = \sum_{y \in Y} w_y p_y(x), \quad \sum_{y \in Y} w_y = 1,$$

где  $p_y(\mathbf{x})$  – функция плотности распределения кластера  $\mathbf{y}$ ,  $w_y$  – неизвестная априорная вероятность появления объектов из кластера  $\mathbf{y}$ .

*Гипотеза о форме кластеров.* Объекты описываются  $n$  числовыми признаками  $f_1(\mathbf{x}), \dots, f_n(\mathbf{x})$ ,  $\mathbf{X} \in \mathbf{R}^n$ . Каждый кластер  $\mathbf{y} \in Y$  описывается  $n$ -мерной гауссовской плотностью  $p_y(\mathbf{x}) = N(\mathbf{x}; \boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y)$  с центром  $\boldsymbol{\mu}_y = (\mu_{y1}, \dots, \mu_{yn})$  и диагональной ковариационной матрицей  $\boldsymbol{\Sigma}_y = \text{diag}(\sigma_{y1}^2, \dots, \sigma_{yn}^2)$ .

На Е-шаге по формуле Байеса вычисляются скрытые переменные  $g_{iy}$ . Значение  $g_{iy}$  равно вероятности того, что объект  $\mathbf{x}_i \in \mathbf{X}^{\ell}$  принадлежит кластеру  $\mathbf{y} \in Y$ . На М-шаге уточняются параметры каждого кластера  $(\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y)$ , при этом существенно используются скрытые переменные  $g_{iy}$ .

## 11.4. Статистические алгоритмы

---

---

### Алгоритм 1.3. Кластеризация с помощью ЕМ-алгоритма

---

1: начальное приближение для всех кластеров  $y \in Y$ :

$$w_y := 1/|Y|;$$

$\mu_y :=$  случайный объект выборки;

$$\sigma_{yj}^2 := \frac{1}{\ell|Y|} \sum_{i=1}^{\ell} (f_j(x_i) - \mu_{yj})^2, \quad j = 1, \dots, n;$$

2: **повторять**

3: Е-шаг (expectation):

$$g_{iy} := \frac{w_y p_y(x_i)}{\sum_{z \in Y} w_z p_z(x_i)}, \quad y \in Y, \quad i = 1, \dots, \ell;$$

4: М-шаг (maximization):

$$w_y := \frac{1}{\ell} \sum_{i=1}^{\ell} g_{iy}, \quad y \in Y;$$

$$\mu_{yj} := \frac{1}{\ell w_y} \sum_{i=1}^{\ell} g_{iy} f_j(x_i), \quad y \in Y, \quad j = 1, \dots, n;$$

$$\sigma_{yj}^2 := \frac{1}{\ell w_y} \sum_{i=1}^{\ell} g_{iy} (f_j(x_i) - \mu_{yj})^2, \quad y \in Y, \quad j = 1, \dots, n;$$

5: Отнести объекты к кластерам по байесовскому решающему правилу:

$$y_i := \arg \max_{y \in Y} g_{iy}, \quad i = 1, \dots, \ell;$$

6: **пока**  $y_i$  не перестанут изменяться;

---

## 11.4. Статистические алгоритмы

---

---

### Алгоритм 1.4. Кластеризация с помощью алгоритма $k$ -средних

---

- 1: сформировать начальное приближение центров всех кластеров  $y \in Y$ :  
 $\mu_y$  — наиболее удалённые друг от друга объекты выборки;
  - 2: **повторять**
  - 3: отнести каждый объект к ближайшему центру (аналог E-шага):  
$$y_i := \arg \min_{y \in Y} \rho(x_i, \mu_y), \quad i = 1, \dots, \ell;$$
  - 4: вычислить новое положение центров (аналог M-шага):  
$$\mu_{yj} := \frac{\sum_{i=1}^{\ell} [y_i = y] f_j(x_i)}{\sum_{i=1}^{\ell} [y_i = y]}, \quad y \in Y, \quad j = 1, \dots, n;$$
  - 5: **пока**  $y_i$  не перестанут изменяться;
- 

Алгоритм  $k$ -means крайне чувствителен к выбору начальных приближений центров. Случайная инициализация центров на шаге 1 может приводить к плохим кластеризациям. Для формирования начального приближения лучше выделить  $k$  наиболее удалённых точек выборки: первые две точки выделяются по максимуму всех попарных расстояний; каждая следующая точка выбирается так, чтобы расстояние от неё до ближайшей уже выделенной было максимально.

## 11.5. Иерархическая кластеризация

---

Иерархические алгоритмы кластеризации, называемые также алгоритмами *таксономии*, строят не одно разбиение выборки на непересекающиеся классы, а систему вложенных разбиений. Результат таксономии обычно представляется в виде таксономического дерева - *дендрограммы*.

- **Дивизимные** или нисходящие алгоритмы разбивают выборку на всё более и более мелкие кластеры.
- **Агломеративные** или восходящие алгоритмы, в которых объекты объединяются во всё более и более крупные кластеры.

Сначала каждый объект считается отдельным кластером. Для одноэлементных кластеров естественным образом определяется функция расстояния

$$R(\{x\}, \{x'\}) = \rho(x, x').$$

Затем запускается процесс слияний. На каждой итерации вместо пары самых близких кластеров **U** и **V** образуется новый кластер **W = U ∪ V**. Расстояние от нового кластера **W** до любого другого кластера **S** вычисляется по расстояниям **R(U, V)**, **R(U, S)** и **R(V, S)**, которые к этому моменту уже должны быть известны:

$$R(U \cup V, S) = \alpha_U R(U, S) + \alpha_V R(V, S) + \beta R(U, V) + \gamma |R(U, S) - R(V, S)|$$

## 11.5. Иерархическая кластеризация

---

*Расстояние ближнего соседа:*

$$R^b(W, S) = \min_{w \in W, s \in S} \rho(w, s); \quad \alpha_U = \alpha_V = \frac{1}{2}, \beta = 0, \gamma = -\frac{1}{2}.$$

*Расстояние дальнего соседа:*

$$R^d(W, S) = \max_{w \in W, s \in S} \rho(w, s); \quad \alpha_U = \alpha_V = \frac{1}{2}, \beta = 0, \gamma = \frac{1}{2}.$$

*Среднее расстояние:*

$$R^c(W, S) = \frac{1}{|W||S|} \sum_{w \in W} \sum_{s \in S} \rho(w, s); \quad \alpha_U = \frac{|U|}{|W|}, \alpha_V = \frac{|V|}{|W|}, \beta = \gamma = 0.$$

*Расстояние между центрами:*

$$R^q(W, S) = \rho^2 \left( \sum_{w \in W} \frac{w}{|W|}, \sum_{s \in S} \frac{s}{|S|} \right); \quad \alpha_U = \frac{|U|}{|W|}, \alpha_V = \frac{|V|}{|W|}, \beta = -\alpha_U \alpha_V, \gamma = 0.$$

*Расстояние Уорда:*

$$R^y(W, S) = \frac{|S||W|}{|S|+|W|} \rho^2 \left( \sum_{w \in W} \frac{w}{|W|}, \sum_{s \in S} \frac{s}{|S|} \right); \quad \alpha_U = \frac{|S|+|U|}{|S|+|W|}, \alpha_V = \frac{|S|+|V|}{|S|+|W|}, \beta = \frac{-|S|}{|S|+|W|}, \gamma = 0.$$

## 11.5. Иерархическая кластеризация

---

---

### Алгоритм 1.5. Агломеративная кластеризация Ланса-Уильямса

---

- 1: инициализировать множество кластеров  $C_1$ :  
 $t := 1; \quad C_t = \{\{x_1\}, \dots, \{x_\ell\}\};$
  - 2: **для всех**  $t = 2, \dots, \ell$  ( $t$  — номер итерации):
  - 3:   найти в  $C_{t-1}$  два ближайших кластера:  
     $(U, V) := \arg \min_{U \neq V} R(U, V);$   
     $R_t := R(U, V);$
  - 4:   изъять кластеры  $U$  и  $V$ , добавить слитый кластер  $W = U \cup V$ :  
     $C_t := C_{t-1} \cup \{W\} \setminus \{U, V\};$
  - 5:   **для всех**  $S \in C_t$
  - 6:     вычислить расстояние  $R(W, S)$  по формуле Ланса-Уильямса;
-

## 11.6. Определение числа кластеров

---

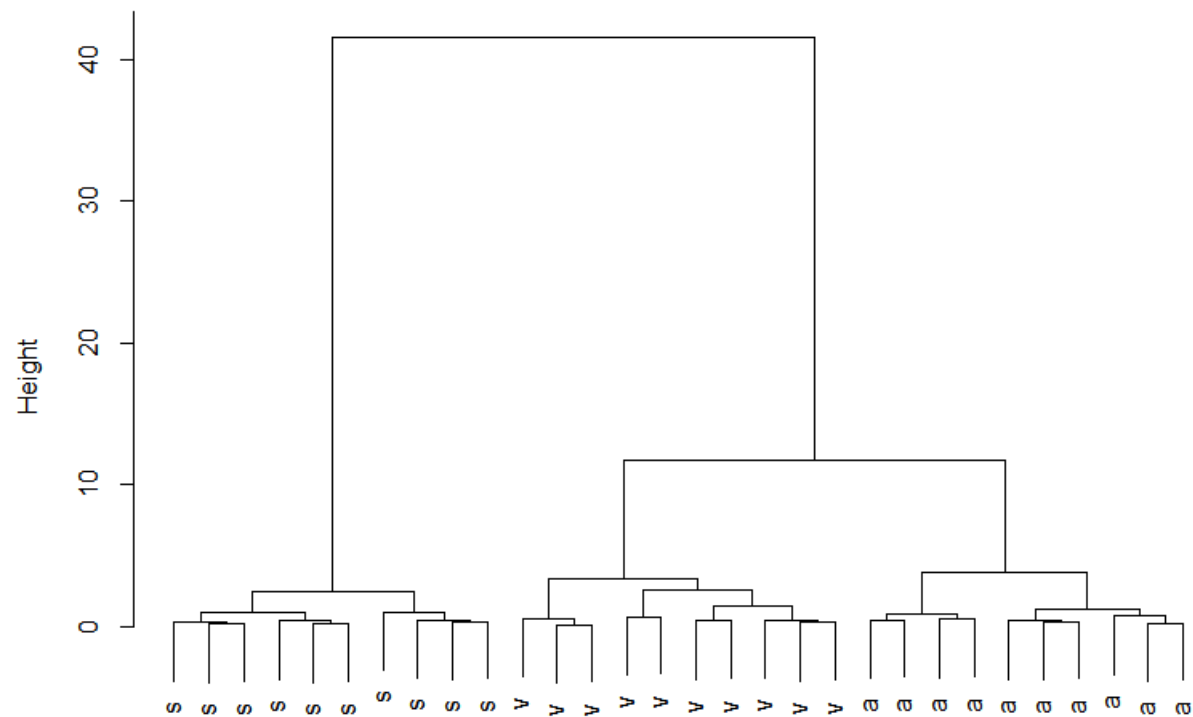
На горизонтальной оси находится интервал максимальной длины  $|R_{t+1}-R_t|$ , и в качестве результирующей кластеризации выдаётся множество кластеров  $C_t$ . Число кластеров равно  $K = \ell - t + 1$ . При необходимости можно задать ограничение на минимальное и максимальное число кластеров  $K_0 \leq K \leq K_1$  и выбирать  $t$ , удовлетворяющие ограничениям  $\ell - K_1 + 1 \leq t \leq \ell - K_0 + 1$ .

Во многих прикладных задачах интерес представляет таксономическое дерево целиком, и определять оптимальное число кластеров не имеет особого смысла.

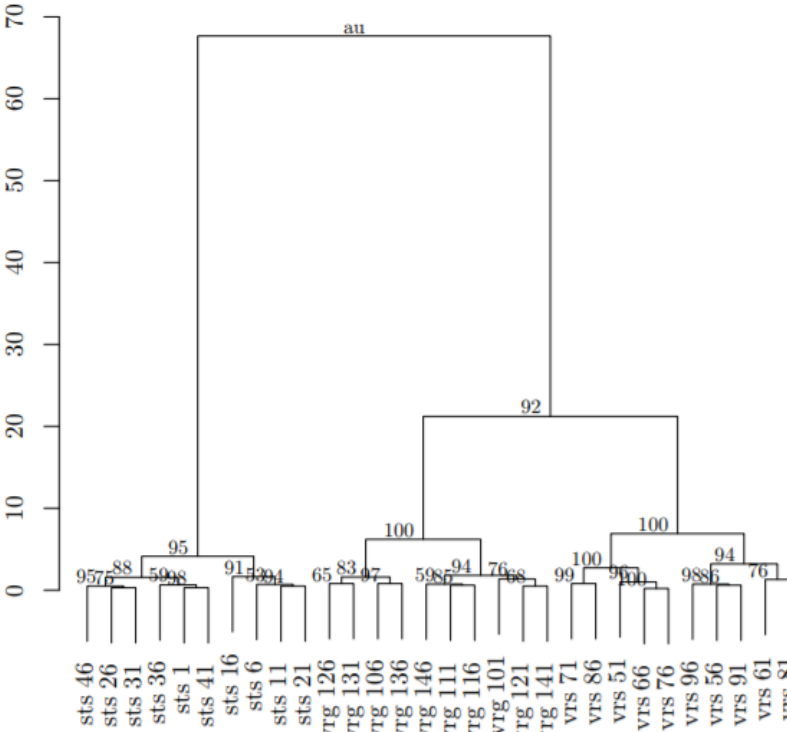
## 11.7. Кластеризация в R

---

```
> library(cluster)
> iriss <- iris[seq(1,nrow(iris),5),]
> iriss.dist <- daisy(iriss[,1:4])
> iriss.h <- hclust(iriss.dist, method="ward")
> plot(iriss.h, labels=abbreviate(iriss[,5],1, method="both.sides"), main="")
```







## 11.7. Кластеризация в R

---

```
> eq <- read.table("data/eq.txt", h=TRUE)
```

```
> eq.k <- kmeans(eq[,-1], 2)
```

```
> table(eq.k$cluster, eq$SPECIES)
```

	arvense	fluviatile
1	37	5
2	1	41

```
> iris.f <- fanny(iris[,1:4], 3)
```

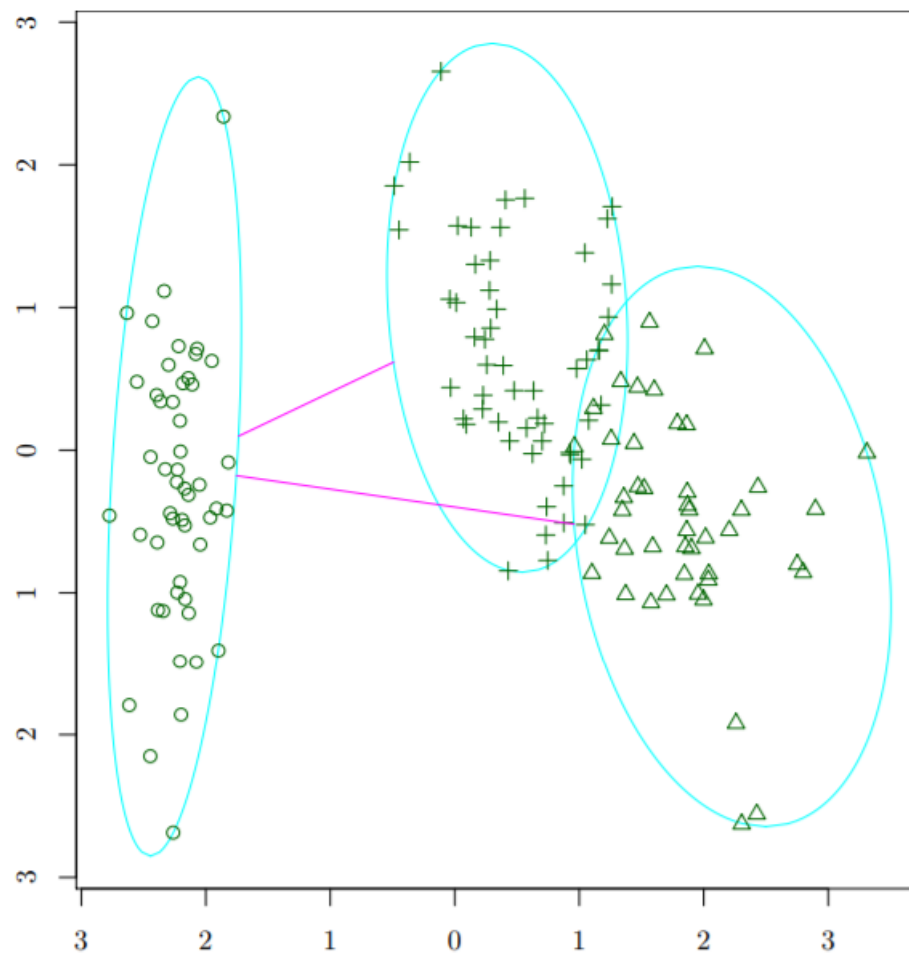
```
> plot(iris.f, which=1, main="")
```

```
> head(data.frame(sp=iris[,5], iris.f$membership))
```

	sp	X1	X2	X3
1	setosa	0.9142273	0.03603116	0.04974153
2	setosa	0.8594576	0.05854637	0.08199602
3	setosa	0.8700857	0.05463714	0.07527719
4	setosa	0.8426296	0.06555926	0.09181118
5	setosa	0.9044503	0.04025288	0.05529687
6	setosa	0.7680227	0.09717445	0.13480286

## 11.7. Кластеризация в R

---



## Литература

---

**Мастицкий С. Э., Шитиков В. К.** (2014) Статистический анализ и визуализация данных с помощью R. - Электронная книга, 400 с

**Шипунов А. Б., Балдин Е. М., Волкова П. А., Коробейников А. И., Назарова С. А., Петров С. В., Суфиянов В. Г.** (2012) Наглядная статистика. Используем R! - М.: ДМК Пресс, 298 с.

**Кабаков Р. К.** (2014) R в действии. Анализ и визуализация данных на языке R Издательство: ДМК Пресс, 580 с.

**Лекции К.В. Воронцова по алгоритмам кластеризации и многомерному шкалированию**