

# **Ординация и классификация с использованием мер сходства-различия**

**Математические методы в зоологии с использованием R**

Марина Варфоломеева

- 1 Коэффициенты сходства и различия**
- 2 Неметрическое многомерное шкалирование**
- 3 Кластерный анализ**
- 4 Методы класстеризации на основании расстояний**
- 5 Сравнение и интерпретация результатов кластеризации**

# Меры сходства и различия, ординация, классификация

## Вы сможете

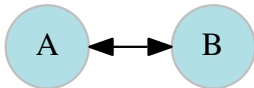
- Выбирать подходящий для данных коэффициент сходства/различия
- Представлять многомерные данные в меньшем числе измерений при помощи неметрического многомерного шкалирования
- Строить дендрограммы при помощи подходящего метода агрегации

## Коэффициенты сходства и различия

# Коэффициенты сходства и различия

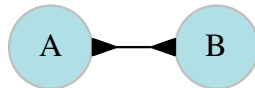
## Расстояния

$$d \geq 0$$



## Сходства

$$0 \leq S \leq 1 \text{ или } -1 \leq S \leq 1$$



- Используются в качестве исходных данных для многих видов многомерных анализов, в т.ч. для неметрического многомерного шкалирования и некоторых видов кластерного анализа
- Из сходств можно получить расстояния и наоборот
- Свои коэффициенты для количественных и качественных признаков

# Свойства коэффициентов сходства-различия

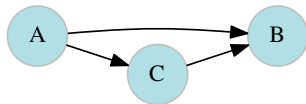
## Метрики и полуметрики

Адекватность:  $d_{A,A} = 0$



## Только метрики

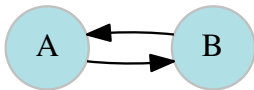
Триангулярность:  $d_{A,B} \leq d_{A,C} + d_{C,B}$



## Неметрики

Все остальное

Симметричность:  $d_{A,B} = d_{B,A}$



# Свойства коэффициентов сходства-различия

## Нестандартные

$$-\inf \leq d \leq \inf$$

## Стандартные

$$d_{min} \leq d \leq d_{max}$$

- частный случай стандартных коэффициентов - коррелятивные коэффициенты сходства

$$-1 \leq S \leq 1$$

# Примеры коэффициентов сходства-различия

## Метрики (расстояния):

- без стандартизации:
  - Евклидово расстояние
  - Манхеттен (расстояние городских кварталов)
- со стандартизацией:
  - Канберра
  - хи-квадрат
  - Евклидово расстояние, рассчитанное по стандартизованным данным

## Неметрики:

- со стандартизацией:
  - коррелятивные:
    - корреляция Пирсона
  - некоррелятивные:
    - коэффициент Брея-Куртиса

## Полуметрики:

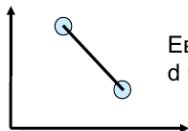
- расстояние Махаланобиса



# Если количественные признаки измерены в одинаковых шкалах

## Метрики без стандартизации

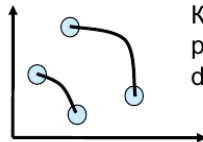
- Евклидово расстояние



Евклидово расстояние  
 $d = [\sum (x_{ik} - x_{jk})^2]^{-1/2}$

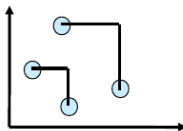
## Неевклидовы метрики

- Квадрат Евклидова расстояния



Квадрат Евклидова расстояния  
 $d = (1/k) \cdot \sum (x_{ik} - x_{jk})^2$

- Манхэттенское расстояние



Манхэттенское расстояние  
 (городские кварталы)  
 $d = 1/k \sum |x_{ik} - x_{jk}|$

## Если количественные признаки измерены в разных шкалах

### Можно стандартизовать исходные данные

- Евклидово (или другое) расстояние, рассчитанное по стандартизованным данным

### Можно использовать коэффициенты со стандартизацией

- Канберра (метрика)  $d = \sum \frac{|x_{ik} - x_{jk}|}{|x_{ik}| + |x_{jk}|}$
- хи-квадрат (метрика)  $\chi^2 = \sqrt{\sum \frac{1}{c_k} (x_{ik} - x_{jk})^2}$
- Коэффициент Махаланобиса (неметрика, not a distance)  $d = \frac{\sum x_{ik} - x_{jk}}{\sigma^2}$
- Корреляция Браве-Пирсона (коррелятивный)  $S = \frac{\sum (x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j)}{n\sigma_i^2\sigma_j^2}$
- Коэффициент Брея-Куртиса (не метрика)  $BC_{ij} = \frac{2C_{ij}}{S_i + S_j}$ , где  $C_{ij}$  - сумма минимальных значений из тех, которые не равны нулю для обоих объектов,  $S_i$  и  $S_j$  - общее число ненулевых значений признаков для обоих объектов.

## Если признаки — подсчеты численности

### Можно стандартизовать исходные данные

Простая стандартизация не подходит (счет, не может быть среднее 0)

Можно использовать трансформации:

- корень, корень 4-й степени
- логарифмирование со сдвигом ( $\log_{10}(x + 1)$ )

### Можно использовать коэффициенты со стандартизацией

- Канберра (метрика)  $d = \sum \frac{|x_{ik} - x_{jk}|}{|x_{ik}| + |x_{jk}|}$
- хи-квадрат (метрика)  $\chi^2 = \sqrt{\sum \frac{1}{c_k} (x_{ik} - x_{jk})^2}$

## Если признаки - доли или проценты

- хи-квадрат (метрика)  $\chi^2 = \sqrt{\sum \frac{1}{c_k} (x_{ik} - x_{jk})^2}$
- коэффициент Брея-Куртиса (не метрика)  $BC_{ij} = \frac{2C_{ij}}{S_i + S_j}$
- Евклидово расстояние  $d = \sqrt{\sum (x_{ik} + x_{jk})^2}$

## Если используются бинарные данные (присутствие-отсутствие признака)

$I \backslash J$	+	-
+	a	b
-	c	d

$I, J$  – множества

$$n_j = a + c \quad n_i = a + b$$

$$n = a + b + c + d$$

	I	J	
1	+	+	a – сходство по наличию
2	+	-	b – различие
3	-	+	c – различие
4	-	-	d – сходство по отсутствию

# Примеры коэффициентов для качественных данных

## Jaccard и Russel Rao

$I \setminus J$	+	-
+	a	b
-	c	d

Jaccard  
 $S = a/(a+b+c)$

Russel, Rao  
 $S = a/n$

С учетом сходства  
по отсутствию

Без учета сходства  
по отсутствию



$a=2, b=1, c=0, d=2$



$a=0, b=1, c=2, d=2$

## Если данные смешанные (качественные и количественные)

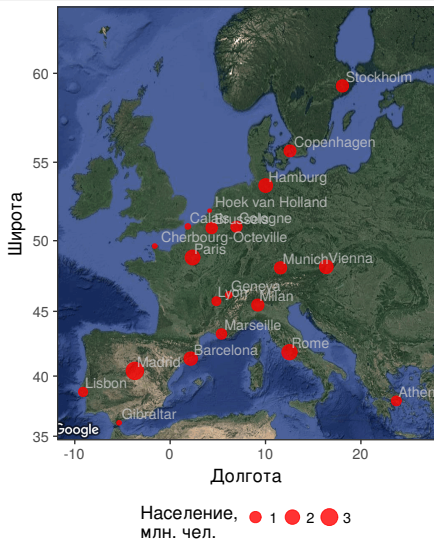
### Коэффициенты для смешанных данных

- расстояние Говера

# Неметрическое многомерное шкалирование



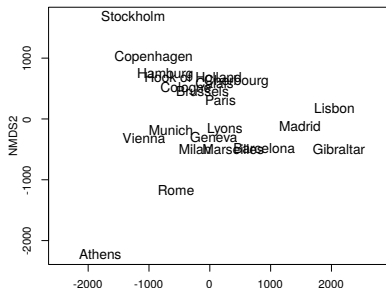
# Неметрическое многомерное шкалирование визуализирует отношения между объектами на основе расстояний между ними



Если бы мы знали расстояния по автодорогам между городами Европы

#	Athens	Barcelona	Brussels	Calais	Cherbourg
# Barcelona	3313				
# Brussels	2963	1318			
# Calais	3175	1326	204		
# Cherbourg	3339	1294	583		

мы бы смогли восстановить по ним карту

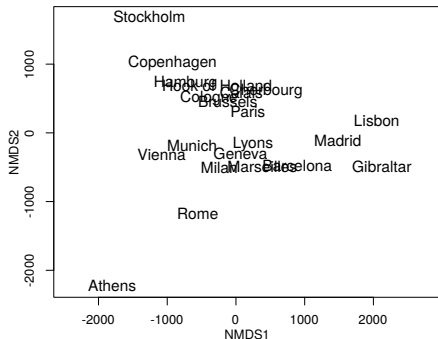


# Неметрическое многомерное шкалирование

Неметрическое многомерное шкалирование (nonmetric multidimensional scaling, nMDS) — метод визуализации отношений между объектами в пространстве с небольшим числом измерений.

Исходные данные — матрица расстояний между объектами в многомерном пространстве

- nMDS подбирает расстояния между объектами на графике так, чтобы сохранились соотношения исходных расстояний между ними. Т.е. если исходно A и B были ближе, чем B и C, то и в результате они должны быть ближе, чем B и C.
- Ординацию nMDS можно поворачивать, отражать, сдвигать - результат от этого не изменится.



## Пример: Морфометрия поссумов



© Hasitha Tudugalle | Photography

Possum by Hasitha Tudugalle on Flickr  
[https://www.flickr.com/photos/hasitha\\_tudugalle/6037880962](https://www.flickr.com/photos/hasitha_tudugalle/6037880962)

Данные Lindenmayer et al. (1995)

## Знакомимся с данными

```
library(DAAG)
data(possum)
colnames(possum)
```

```
# [1] "case"      "site"      "Pop"       "sex"       "age"       "hdlngth"
# [7] "skullw"    "totlngth" "taill"     "footlght" "earconch" "eye"
# [13] "chest"     "belly"
```

```
sum(is.na(possum))
```

```
# [1] 3
```

```
possum[!complete.cases(possum), ]
```

```
#      case site Pop sex age hdlngth skullw totlngth taill footlght
# BB36   41    2 Vic  f  5   88.4   57.0         83  36.5      NA
# BB41   44    2 Vic  m  NA   85.1   51.5         76  35.5     70.3
# BB45   46    2 Vic  m  NA   91.4   54.4         84  35.0     72.8
#      earconch eye chest belly
# BB36     40.3 15.9  27.0  30.5
# BB41     52.6 14.4  23.0  27.0
```

```
# Добавим названия сайтов
possum$site <- factor(possum$site, levels = 1:7,
  labels = c("Cambarville", "Bellbird",
    "Whian Whian", "Byranger",
    "Conondale ", "Allyn River",
    "Bulburin"))
```

Отберем переменные, с которыми будем работать

```
colnames(possum)
```

```
# [1] "case"      "site"      "Pop"       "sex"       "age"       "hdlngth"
# [7] "skullw"    "totlngth"  "taill"     "footlgth"  "earconch"  "eye"
# [13] "chest"     "belly"
```

```
possumc <- possum[complete.cases(possum), c(3:14)]
```

## Неметрическое многомерное шкалирование

Построим ординацию поссумов на основе их сходства по морфометрии и возрасту.

Функция `metaMDS` много раз итеративно подбирает координаты поссумов в новом пространстве (двумерном по умолчанию) и сохраняет лучшую конфигурацию.

Поскольку по умолчанию `autotransform = TRUE`, данные предварительно подвергаются двойной ("висконсинской") стандартизации (см. `?metaMDS`).

**Если у вас не данные о сообществах, то это нужно отключить**, но пока мы продолжим с трансформацией.

```
library(vegan)
ord_euclid_wisk <- metaMDS(possumc[, 3:10], distance = "euclid", autotransform = TRUE)

# Square root transformation
# Wisconsin double standardization
# Run 0 stress 0.03822067
# Run 1 stress 0.04440752
# Run 2 stress 0.03822714
# ... Procrustes: rmse 0.0005365211 max resid 0.003930073
# ... Similar to previous best
# Run 3 stress 0.04030862
# Run 4 stress 0.04367754
# Run 5 stress 0.04620427
```

## Качество подгонки модели

**stress** - оценивает, насколько были искажены исходные расстояния между объектами при снижении размерности

```
ord_euclid_wisk$stress
```

```
# [1] 0.03822067
```

- Эмпирическое правило предложено Краскалом:
  - $S > 20\%$  плохо
  - $S = 10\%$  нормально
  - $S < 5\%$  хорошо
  - $S = 0$  прекрасно

# Ординация

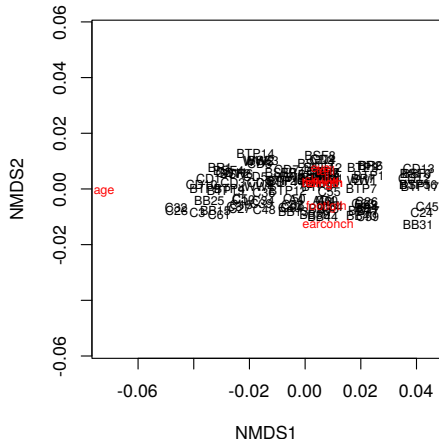
Координаты наблюдений:

```
head(ord_euclid_wisk$points, 10)
```

#		MDS1	MDS2
# C3	-0.03856702	-0.008631770	
# C5	-0.02326647	-0.003786437	
# C10	-0.02192530	-0.003824108	
# C15	-0.02278328	-0.005940789	
# C23	0.02197540	-0.005716512	
# C24	0.04206661	-0.008601634	
# C26	0.02220393	-0.004639378	
# C27	-0.02330802	-0.007341595	
# C28	-0.04597908	-0.007863745	
# C31	-0.02401606	-0.006624305	

График ординации:

```
ordiplot(ord_euclid_wisk, type = "t")
```





## Задание:

При помощи `ggplot2` постройте график неметрического многомерного шкалирования.

Для графика используйте координаты точек `ord_euclid_wisk$points` и исходные данные.

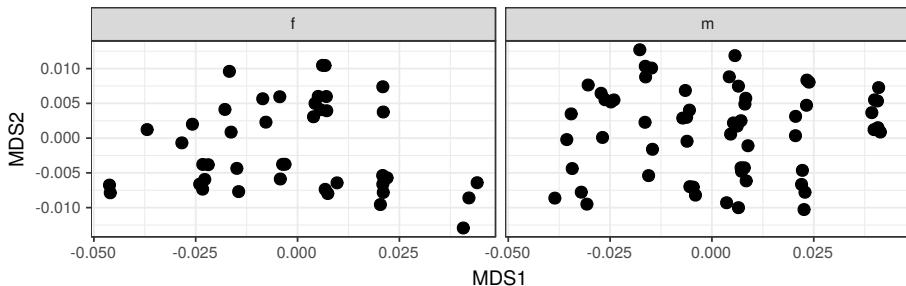
Раскрасьте график по значениям переменных `Pop` и `age`

Изобразите поссумов разного пола на разных панелях

## Решение:

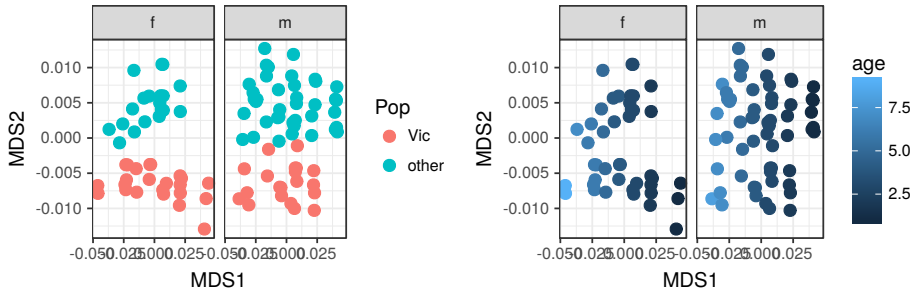
```
library(ggplot2)
theme_set(theme_bw(base_size = 12) + theme(legend.key = element_blank()))

ord_euclid_wisk_points <- data.frame(ord_euclid_wisk$points, possumc)
gg <- ggplot(ord_euclid_wisk_points, aes(x = MDS1, y = MDS2)) +
  geom_point() + facet_wrap(~sex) +
  theme(legend.key = element_blank())
gg
```



## Решение: раскрашенные графики

```
library(gridExtra)
grid.arrange(gg + aes(colour = Pop),
             gg + aes(colour = age), ncol = 2)
```



## Задание:

Постройте nMDS ординацию при помощи евклидова расстояния, **без стандартизации**

Воспользуйтесь справкой к функции `metaMDS()`, чтобы узнать, какие аргументы потребуется изменить.

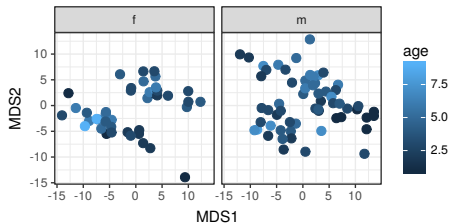
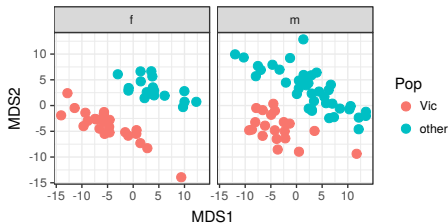
## Решение:

```
ord_raw <- metaMDS(possumc[, 3:10], dis = "euclidean", autotransform = FALSE)
```

```
# Run 0 stress 0.1008672
# Run 1 stress 0.1180912
# Run 2 stress 0.1180945
# Run 3 stress 0.1008672
# ... Procrustes: rmse 0.0000224498 max resid 0.0001458249
# ... Similar to previous best
# Run 4 stress 0.1008672
# ... New best solution
# ... Procrustes: rmse 0.0000146035 max resid 0.00008753054
# ... Similar to previous best
# Run 5 stress 0.1008672
# ... Procrustes: rmse 0.00001884783 max resid 0.0001348232
# ... Similar to previous best
# Run 6 stress 0.1008672
# ... Procrustes: rmse 0.000008849588 max resid 0.0000574325
# ... Similar to previous best
# Run 7 stress 0.1297469
# Run 8 stress 0.1008672
# ... New best solution
# ... Procrustes: rmse 0.000003960715 max resid 0.00002666474
# ... Similar to previous best
# Run 9 stress 0.1008672
# ... Procrustes: rmse 0.000003386066 max resid 0.00001736033
```

# Графики ординации по матрице евклидовых расстояний без стандартизации

```
ord_raw_points <- data.frame(ord_raw$points, possumc)
library(gridExtra)
grid.arrange(gg %>% ord_raw_points + aes(colour = Pop),
gg %>% ord_raw_points + aes(colour = age),
ncol = 2)
```



# Как изменилась сама ординация?

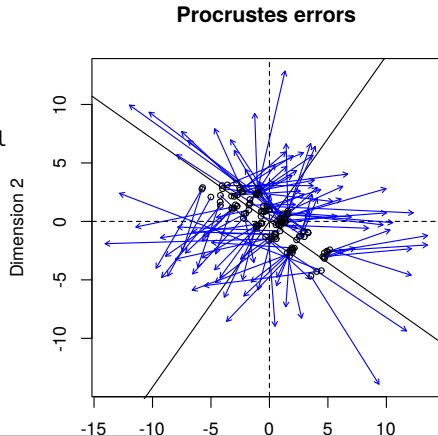
## Прокрустово преобразование

procrustes(что\_стало, что\_было)

```
proc <- procrustes(ord_raw,
                   ord_euclid_wisk)
proc
```

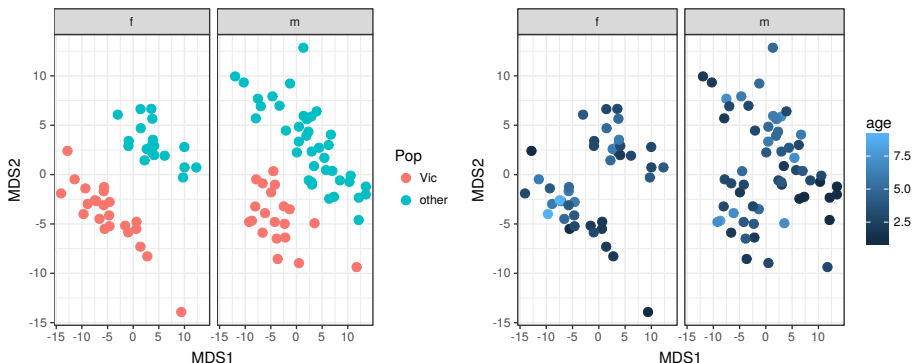
```
plot(proc)
```

```
#
# Call:
# procrustes(X = ord_raw, Y = ord_eucl
#
# Procrustes sum of squares:
# 5670
```



**Похоже, что в этом случае лучшая ординация была получена при использовании евклидова расстояния без висконсинской стандартизации**

```
grid.arrange(gg %+% ord_raw_points + aes(colour = Pop),
gg %+% ord_raw_points + aes(colour = age),
ncol = 2)
```





# Кластерный анализ

## Пример: поссумы

Морфометрия самок поссумов

```
library(DAAG)
data(fossum)
# создадим "говорящие" имена строк
rownames(fossum) <- paste(fossum$Pop, rownames(fossum), sep = "_")
fossumc <- fossum[complete.cases(fossum), 5:14]
```

# Какие бывают методы построения деревьев?

## Методы кластеризации на основании расстояний (о них сегодня)

- Метод ближайшего соседа
- Метод отдаленного соседа
- Метод среднегруппового расстояния
- Метод Варда
- и т.д. и т.п.

## Методы кластеризации на основании признаков

- Метод максимальной бережливости
- Метод максимального правдоподобия

## Методы кластеризации на основании расстояний

## Этапы кластеризации

## От чего зависит результат кластеризации

Результат кластеризации зависит от

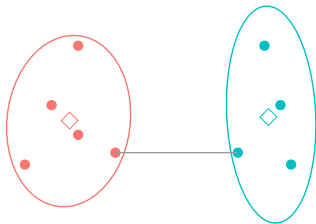
- коэффициента сходства-различия
- от алгоритма кластеризации

```
d <- dist(x = fossumc, method = "euclidean")
```

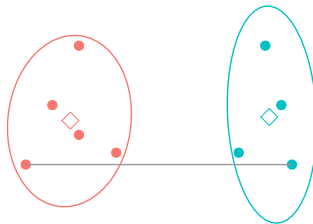
Давайте построим деревья при помощи нескольких алгоритмов кластеризации и сравним их.

# Методы кластеризации

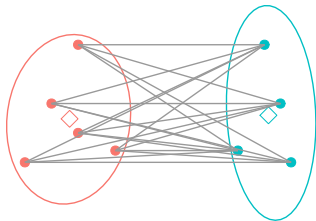
Метод ближайшего соседа



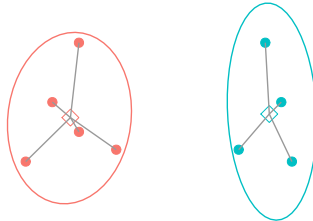
Метод отдаленного соседа



Метод среднегруппового расстояния

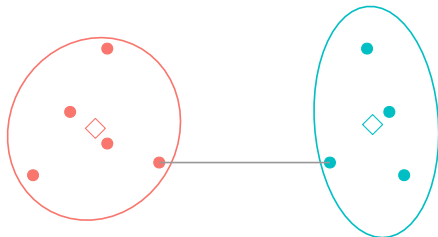


Метод Варда



## Метод ближайшего соседа

- = nearest neighbour = single linkage
- к кластеру присоединяется ближайший к нему кластер/объект
- кластеры объединяются в один на расстоянии, которое равно расстоянию между ближайшими объектами этих кластеров



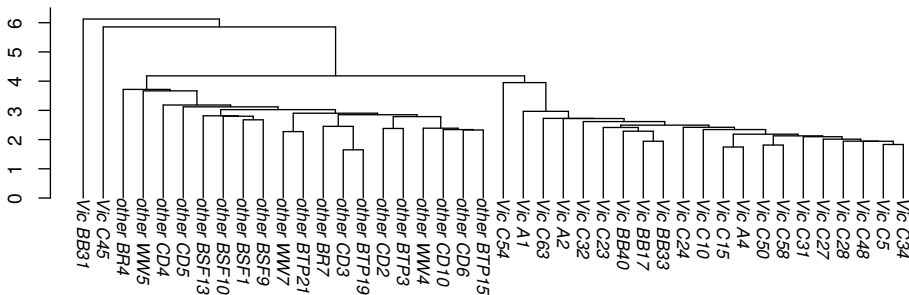
### Особенности:

- Может быть сложно интерпретировать, если нужны группы
- объекты на дендрограмме часто не образуют четко разделенных групп
- часто получаются цепочки кластеров (объекты присоединяются как бы по-одному)
- Хорош для выявления градиентов



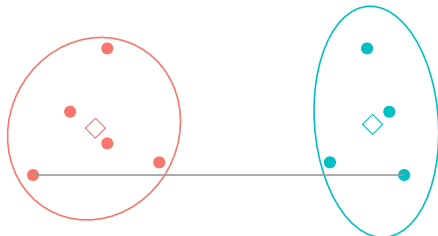
## Метод ближайшего соседа в R

```
hc_single <- hclust(d, method = "single")
library(ape)
ph_single <- as.phylo(hc_single)
par(cex = 0.9) # уменьшаем размер шрифта
plot(ph_single, type = "phylogram", direction = "downwards")
axisPhylo(side = 2)
```



## Метод отдаленного соседа

- = furthest neighbour = complete linkage
- к кластеру присоединяется отдаленный кластер/объект
- кластеры объединяются в один на расстоянии, которое равно расстоянию между самыми отдаленными объектами этих кластеров (следствие - чем более крупная группа, тем сложнее к ней присоединиться)

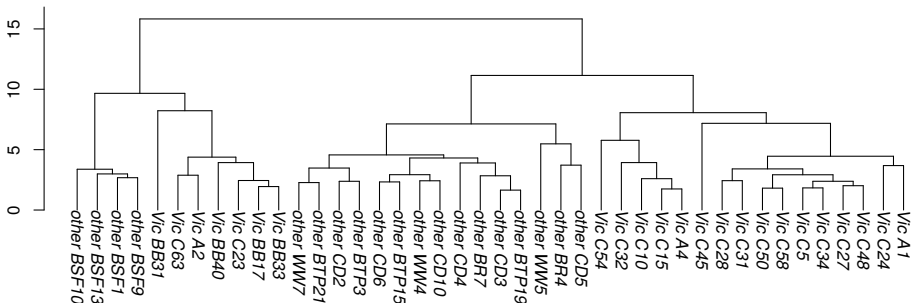


### Особенности:

- На дендрограмме образуется много отдельных некрупных групп
- Хорош для поиска дискретных групп в данных

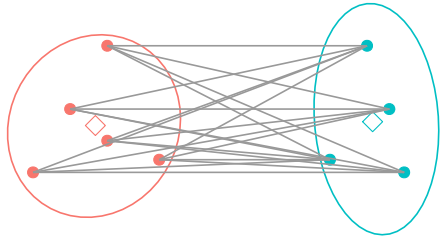
## Метод отдаленного соседа в R

```
ph_compl <- as.phylo(hclust(d, method = "complete"))
plot(ph_compl, type = "phylogram", direction = "downwards")
axisPhylo(side = 2)
```



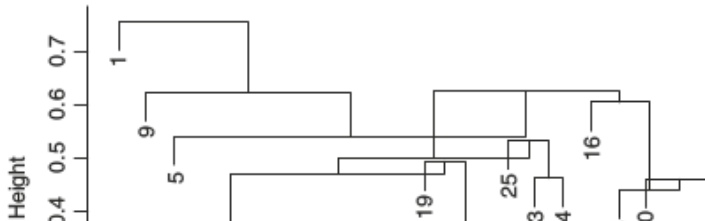
## Метод невзвешенного попарного среднего

- = UPGMA = Unweighted Pair Group Method with Arithmetic mean
- кластеры объединяются в один на расстоянии, которое равно среднему значению всех возможных расстояний между объектами из разных кластеров.



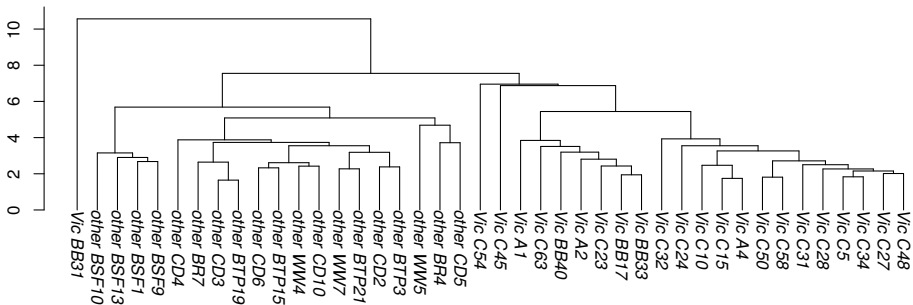
### Особенности:

- UPGMA и WUPGMC иногда могут приводить к инверсиям на дендрограммах



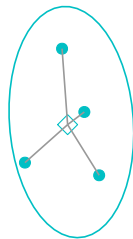
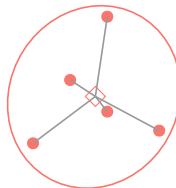
## Метод невзвешенного попарного среднего в R

```
ph_avg <- as.phylo(hclust(d, method = "average"))
plot(ph_avg, type = "phylogram", direction = "downwards")
axisPhylo(side = 2)
```



# Метод Варда

- = Ward's Minimum Variance Clustering
- объекты объединяются в кластеры так, чтобы внутригрупповая дисперсия расстояний была минимальной

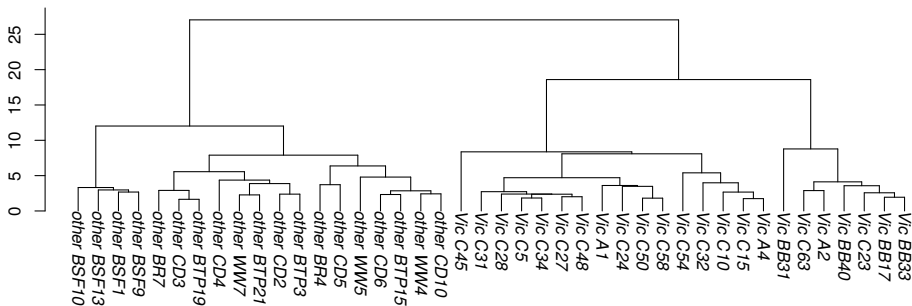


## Особенности:

- метод годится и для неевклидовых расстояний несмотря на то, что внутригрупповая дисперсия расстояний рассчитывается так, как будто это евклидовы расстояния

## Метод Варда в R

```
ph_w2 <- as.phylo(hclust(d, method = "ward.D2"))
plot(ph_w2, type = "phylogram", direction = "downwards")
axisPhylo(side = 2)
```



## **Сравнение и интерпретация результатов кластеризации**



## Кофенетическая корреляция

Кофенетическое расстояние - расстояние между объектами на дендрограмме

Кофенетическую корреляцию можно рассчитать как пирсоновскую корреляцию (обычную) между матрицами исходных и кофенетических расстояний между всеми парами объектов

Метод, который дает наибольшую кофенетическую корреляцию дает кластеры лучше всего отражающие исходные данные

## Кофенетическая корреляция в R

```
c_single <- cophenetic(ph_single)
c_compl <- cophenetic(ph_compl)
c_avg <- cophenetic(ph_avg)
c_w2 <- cophenetic(ph_w2)
```

```
cor(d, as.dist(c_single))
```

```
# [1] 0.7328872
```

```
cor(d, as.dist(c_compl))
```

```
# [1] 0.5999495
```

```
cor(d, as.dist(c_avg)) # лучше всех отражает структуру данных
```

```
# [1] 0.8014266
```

```
cor(d, as.dist(c_w2))
```

```
# [1] 0.6830621
```

## На каком уровне нужно делить дендрограмму на кластеры?

- Можно субъективно, на любом выбранном уровне. Главное, чтобы кластеры были осмысленными и интерпретируемыми.
- Можно выбрать, глядя на распределение расстояний ветвления
- Можно оценить вероятность разделения на кластеры при помощи бутстрепа

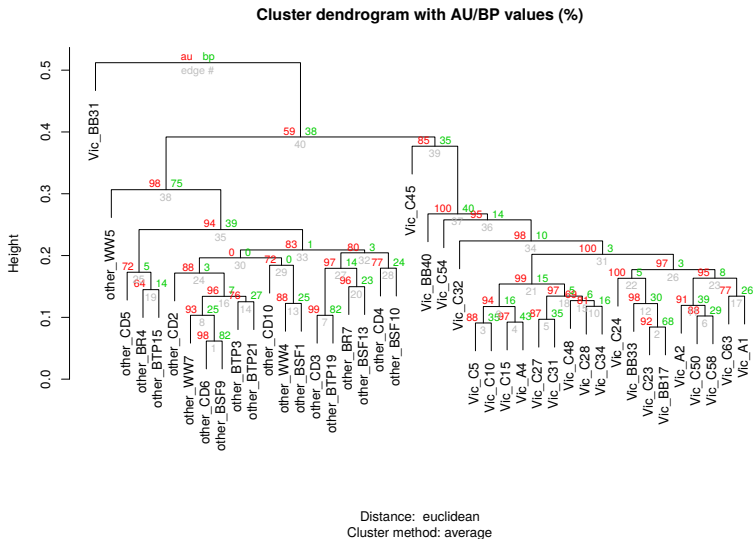
## Бутстреп

```
library(pvclust)
```

```
# итераций должно быть 10000 и больше
# здесь мало для скорости
set.seed(42)
system.time({
  cl_boot <- pvclust(scale(t(fossumc)), method.hclust = "average",
                    nboot = 1000, method.dist = "euclidean")
})
```

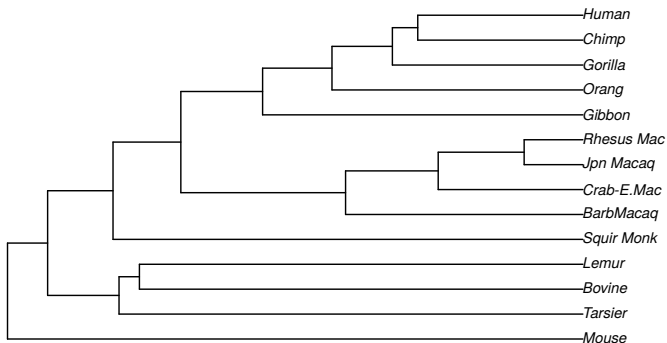
```
# Bootstrap (r = 0.5)... Done.
# Bootstrap (r = 0.6)... Done.
# Bootstrap (r = 0.7)... Done.
# Bootstrap (r = 0.8)... Done.
# Bootstrap (r = 0.9)... Done.
# Bootstrap (r = 1.0)... Done.
# Bootstrap (r = 1.1)... Done.
# Bootstrap (r = 1.2)... Done.
# Bootstrap (r = 1.3)... Done.
# Bootstrap (r = 1.4)... Done.
```

```
plot(cl_boot)
```



## И небольшая демонстрация — дерево по генетическим данным

```
webpage <- "http://evolution.genetics.washington.edu/book/primates.dna"
primates.dna <- read.dna(webpage)
d_pri <- dist.dna(primates.dna, model = "K80")
hc_pri <- hclust(d_pri, method = "average")
ph_pri <- as.phylo(hc_pri)
plot(ph_pri)
axisPhylo()
```



## Take home messages

- Неметрическое многомерное шкалирование - способ снижения размерности, сохраняющий ранги расстояний между объектами
- Направления на графике многомерного шкалирования можно интерпретировать произвольным образом в зависимости от изменения других переменных (не обязательно вдоль осей)
- Результат многомерного шкалирования зависит от выбора коэффициента различия

## Дополнительные ресурсы

- Borcard, D., Gillet, F., Legendre, P., 2011. Numerical ecology with R. Springer.
- Legendre, P., Legendre, L., 2012. Numerical ecology. Elsevier.
- Oksanen, J., 2011. Multivariate analysis of ecological communities in R: vegan tutorial. R package version 2-0.
- Quinn, G.G.P., Keough, M.J., 2002. Experimental design and data analysis for biologists. Cambridge University Press.

Borcard, D., Gillet, F., Legendre, P., 2011. Numerical ecology with R. Springer.

Как работает UPGMA можно посмотреть здесь:

- <http://www.southampton.ac.uk/~relu06/teaching/upgma/>
- pvclust: An R package for hierarchical clustering with p-values [WWW Document], n.d. URL <http://www.sigmath.es.osaka-u.ac.jp/shimo-lab/prog/pvclust/> (accessed 11.7.14).

Для анализа молекулярных данных:

- Paradis, E., 2011. Analysis of Phylogenetics and Evolution with R. Springer.