Анализ главных компонент

Математические методы в зоологии с использованием R

Марина Варфоломеева

Знакомимся с ординацией на примере метода главных компонент

Вы сможете

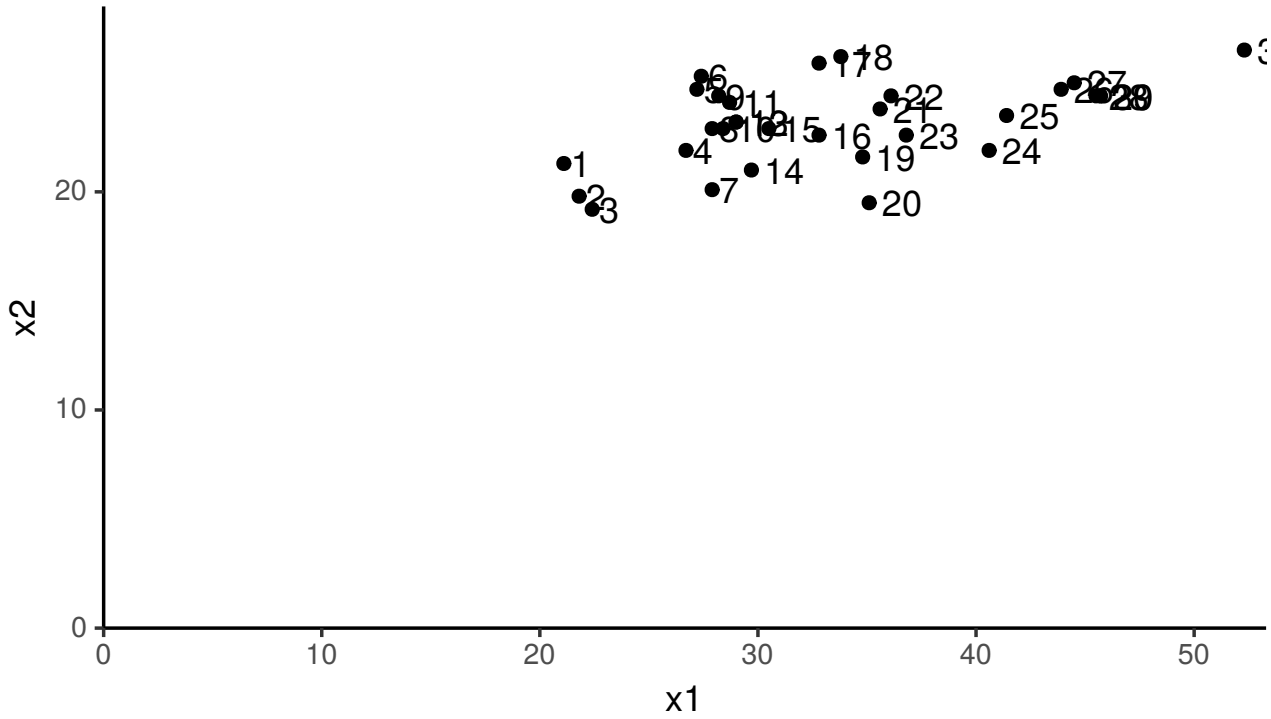
- Проводить анализ главных компонент
- Снижать размерность данных, отбирая меньшее число главных компонент
- Оценивать долю объясненной изменчивости
- Интерпретировать компоненты по значениям факторных нагрузок
- Строить ординацию объектов в пространстве главных компонент
- Извлекать значения факторов объектов для дальнейшего использования с другими видами анализов

Снижение размерности многомерных данных

Анализ главных компонент — способ снижения размерности

Многомерные исходные данные

```
#         x1   x2
# [1,] 21.1 21.3
# [2,] 21.8 19.8
# [3,] 22.4 19.2
# [4,] 26.7 21.9
# [5,] 27.2 24.7
# [6,] 27.4 25.3
```



В этом примере для простоты используются двумерные данные, т.е. у каждого наблюдения (строки) есть два свойства (столбцы). Например, это могут быть свойства деревьев: x1 — диаметр ствола, x2 — высота ствола.

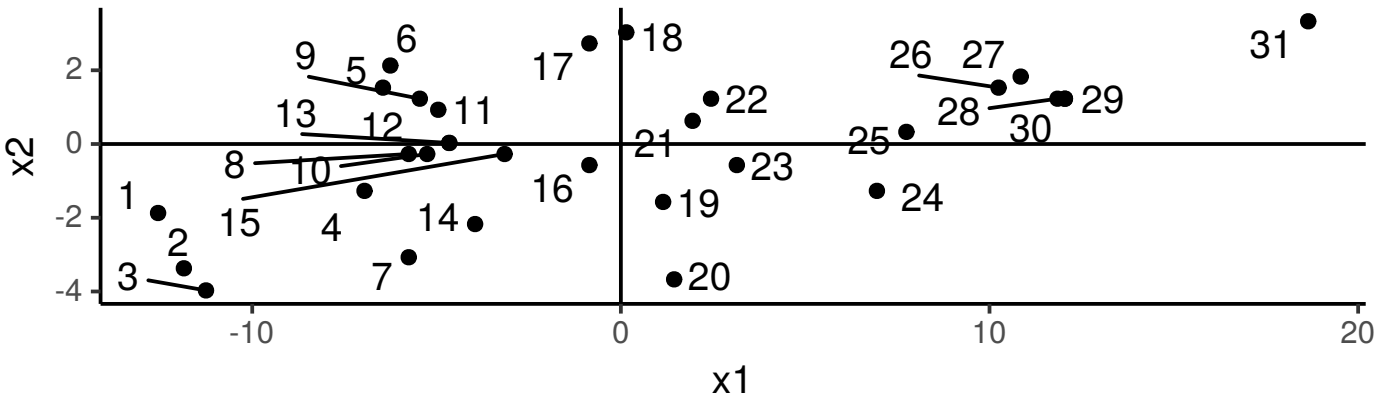
Центрирование

Центрирование

Из каждого значения переменной нужно вычесть среднее значение этой переменной.

Центрированные данные:

```
#          x1   x2
# [1,] -12.6 -1.9
# [2,] -11.9 -3.4
# [3,] -11.3 -4.0
# [4,]  -7.0 -1.3
# [5,]  -6.5  1.5
# [6,]  -6.3  2.1
```



Если центрировать данные (вычесть среднее x1 = 33.7, среднее x2 = 23.2), то центр координат переместится в точку $(\bar{x}_1, \bar{x}_2)$

Матрица ковариаций между признаками

Исходные данные:

```
#         x1   x2
# [1,] 21.1 21.3
# [2,] 21.8 19.8
# [3,] 22.4 19.2
# [4,] 26.7 21.9
# [5,] 27.2 24.7
# [6,] 27.4 25.3
```

Из исходных данных получают матрицу ковариаций:

```
#             x1       x2
# x1 63.484559 8.056645
# x2  8.056645 3.800129
```

Матрица ковариаций

- описывает совместное варьирование нескольких переменных
- по диагонали — дисперсии признаков
- выше и ниже диагонали — ковариации признаков друг с другом

Матрицу ковариаций можно представить в виде собственных векторов и собственных чисел

Матрица ковариаций

```
#           x1       x2
# x1 63.484559 8.056645
# x2  8.056645 3.800129
```

Собственные числа

- используются для оценки вклада главных компонент в общую изменчивость
- дисперсия вдоль собственных векторов пропорциональна их собственным числам

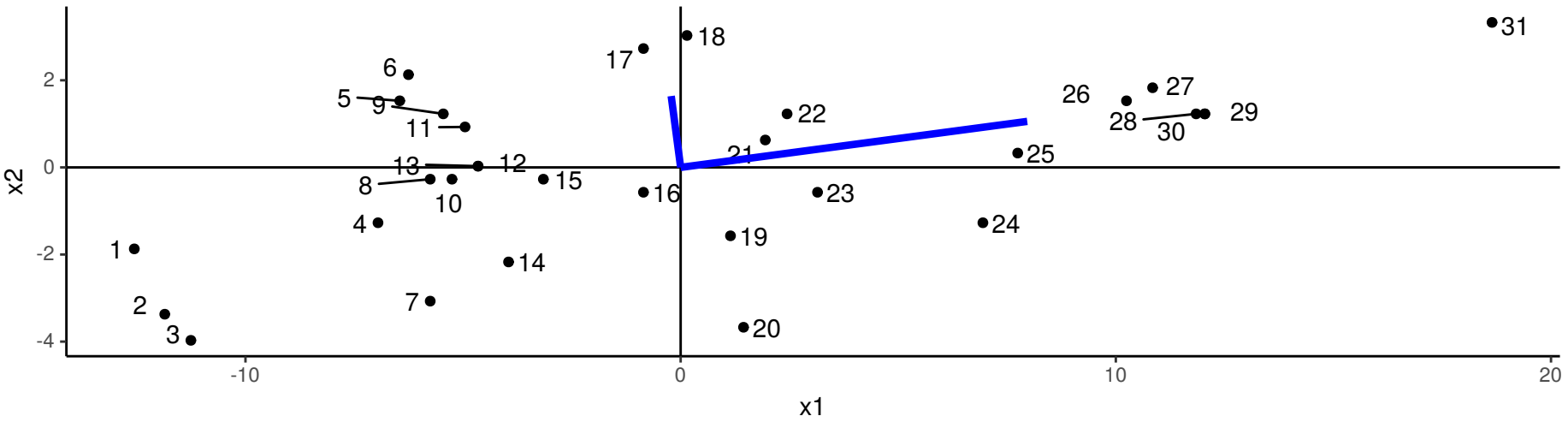
```
# [1] 64.552979  2.731709
```

Собственные векторы

- их столько же, сколько исходных переменных
- перпендикулярны друг другу
- задают направление осей главных компонент
- вдоль первого — максимальная дисперсия данных, вдоль следующего — максимальная дисперсия из оставшейся и т.д.

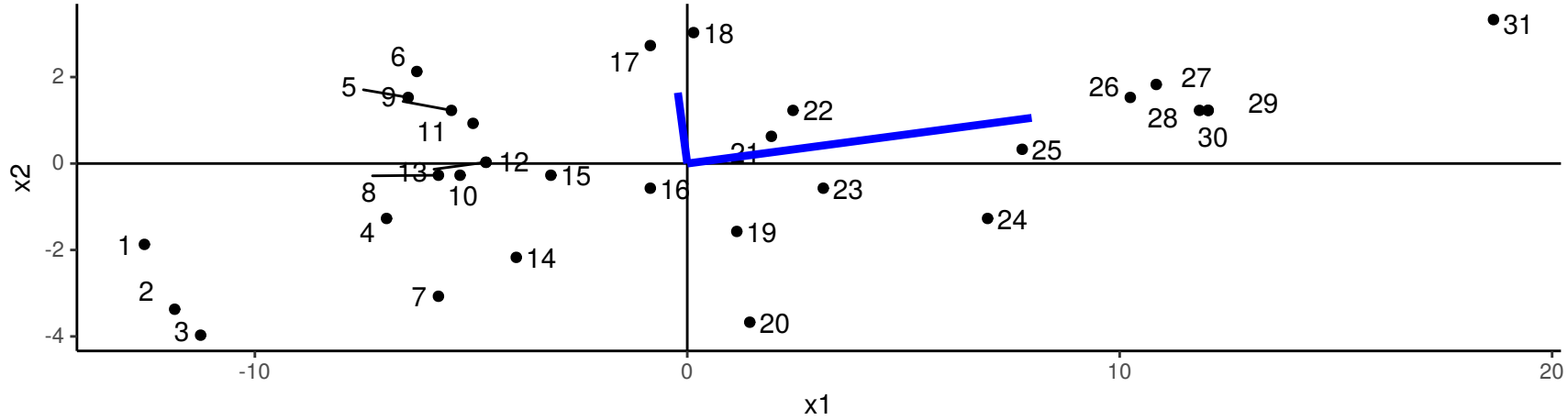
```
#             [,1]       [,2]
# [1,] -0.9913211  0.1314625
# [2,] -0.1314625 -0.9913211
```

С помощью собственных векторов и собственных чисел можно найти в пространстве признаков новые оси, вдоль которых будет максимальный разброс точек.

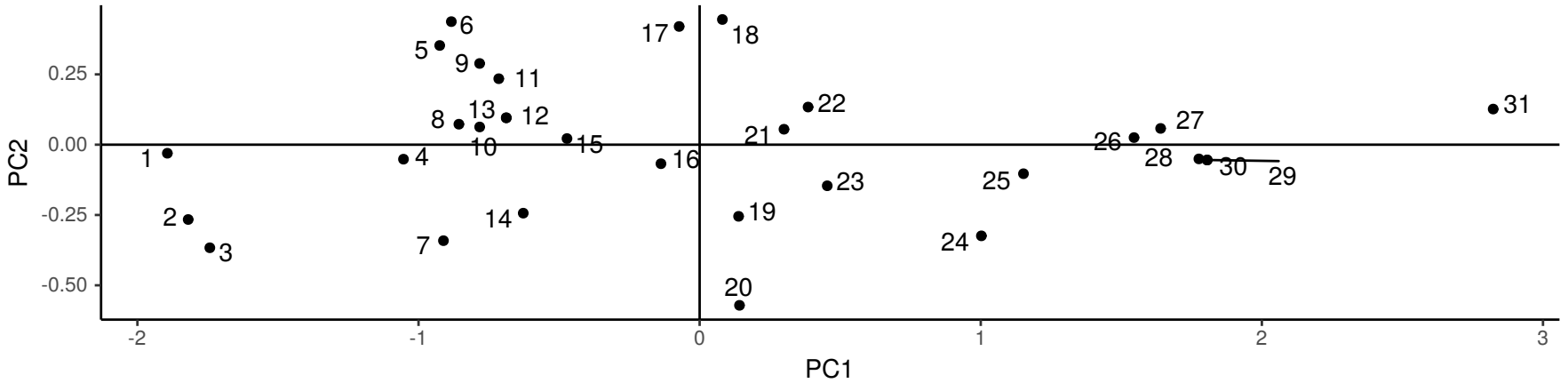


Можно найти новые координаты точек в получившемся новом пространстве

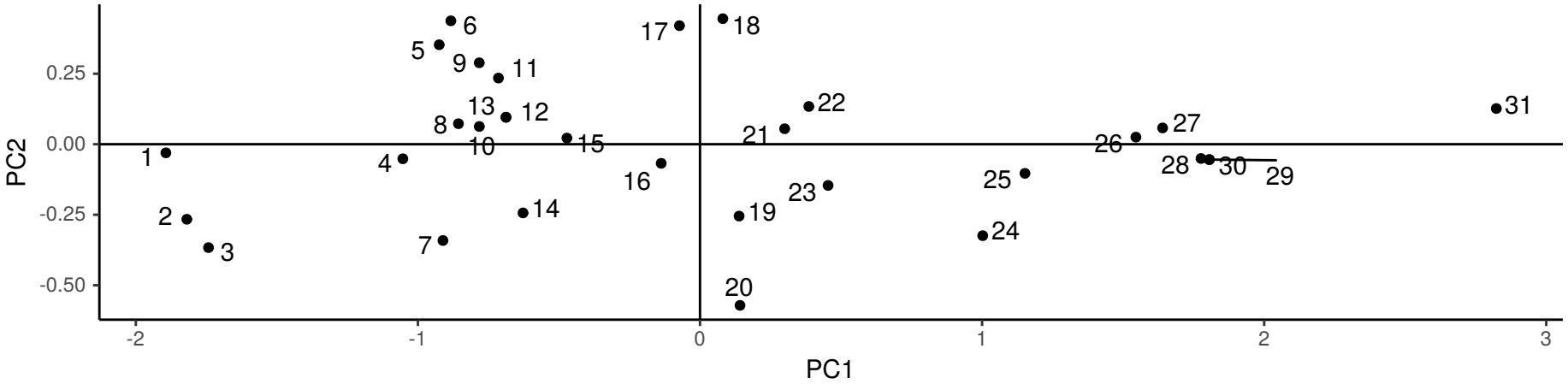
До PCA: объекты и оси в пространстве исходных признаков



После PCA: Объекты в пространстве новых осей (главных компонент)



На графике ординации изображено новое пространство



По собственным числам судим о доле изменчивости, объясненной новыми направлениями осей (компонентами)

- PC1 — больше всего изменчивости
- PC2 — то, что осталось

По новым координатам судим о близости объектов

По факторным нагрузкам исходных переменных на компоненты интерпретируем новые направления

Анализ главных компонент в R

Пример: Морфометрия поссумов

pos by Hasitha Tudugalle on Flickr
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"    "footlgth" "earconch" "eye"
# [13] "chest"    "belly"
```

```
colSums(is.na(possum))
```

```
#     case     site      Pop      sex      age  hdlngth   skullw
#        0        0        0        0        2        0        0
# totlngth    taill footlgth earconch      eye    chest    belly
#        0        0        1        0        0        0        0
```

```
# оставим только строки с полными наблюдениями
pos <- possum[complete.cases(possum), ]
```

```r
# поссумы из разных сайтов из 2 популяций
table(pos$site, pos$Pop)
```

```
#
#     Vic other
#   1  33     0
#   2  10     0
#   3   0     7
#   4   0     7
#   5   0    13
#   6   0    13
#   7   0    18
```

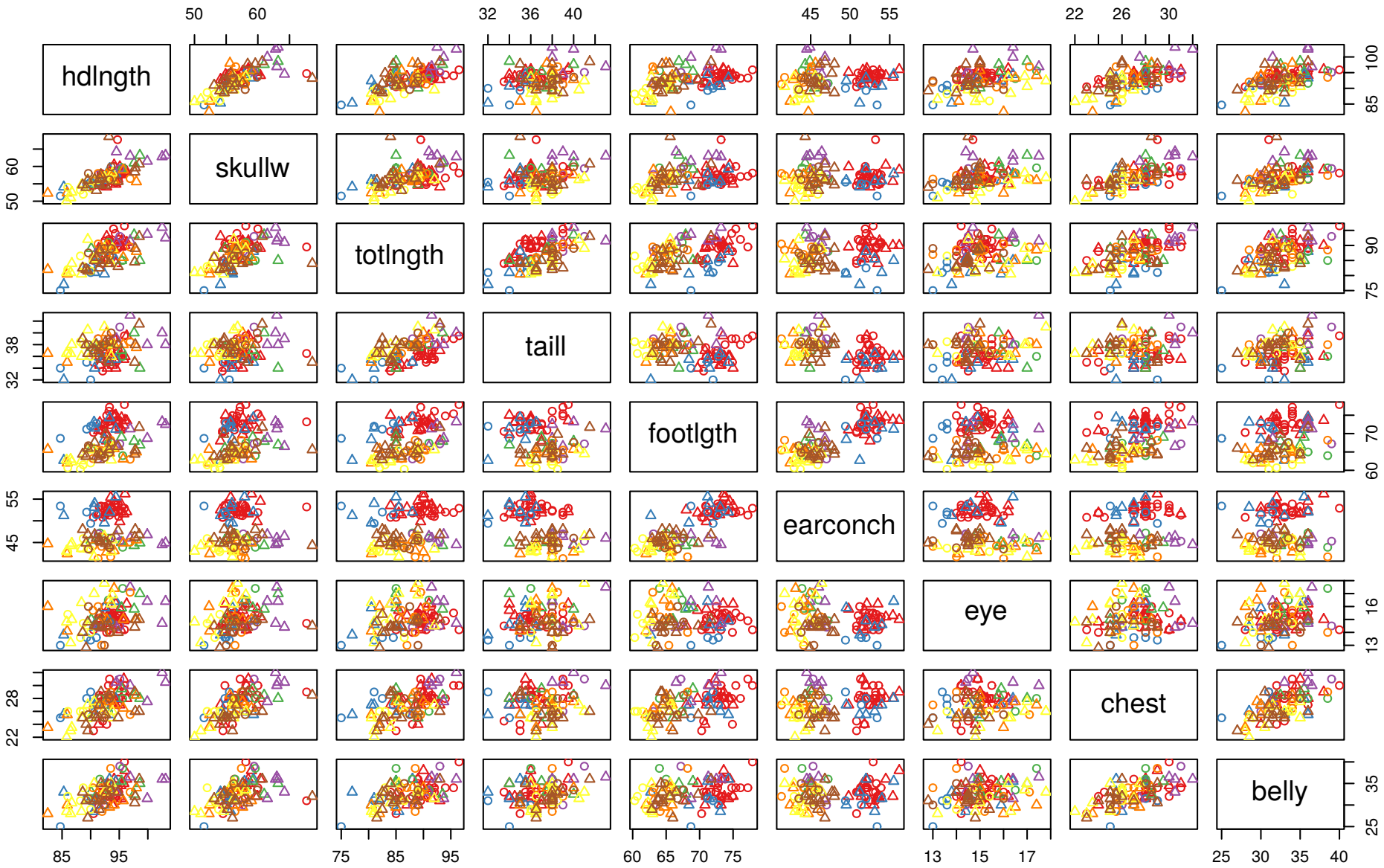
```r
# половой состав выборок из разных сайтов
table(pos$sex, pos$site, pos$Pop)
```

```
# , ,  = Vic
#
#
#      1  2  3  4  5  6  7
#   f 19  4  0  0  0  0  0
#   m 14  6  0  0  0  0  0
#
# , ,  = other
#
#
#      1  2  3  4  5  6  7
#   f  0  0  3  2  6  4  4
#   m  0  0  4  5  7  9 14
```

Как связаны признаки между собой?

Можно построить серию графиков с признаками во всех возможных комбинациях.

```
# сколько всего сайтов
n_sites <- length(unique(pos$site))
# цвета из Брюеровской палитры 'Set1'
library(RColorBrewer)
cols <- brewer.pal(n = n_sites, name = 'Set1')
# график морфометрических переменных
pairs(pos[, 6:14], col = cols[pos$site],
      pch =  as.numeric(pos$sex))
```


hdlngth
skullw
totlngth
taill
footlgth
earconch
eye
chest
belly
50 60
32 36 40
45 50 55
22 26 30
85 100
50 60
75 90
32 38
60 70
45 55
13 16
22 28
25 35
85 95
75 85 95
60 65 70 75
13 15 17
25 30 35 40

Анализ главных компонент

```
library(vegan)
# ординация, используем морфометрические переменные (с hdlngth по belly)
ord <- rda(pos[, 6:14], scale = TRUE)
```

```
summary(ord)
```

Все результаты можно посмотреть при помощи функции summary()

```
#
# Call:
# rda(X = pos[, 6:14], scale = TRUE)
#
# Partitioning of correlations:
#               Inertia Proportion
# Total               9          1
# Unconstrained       9          1
#
# Eigenvalues, and their contribution to the correlations
#
# Importance of components:
#                            PC1    PC2    PC3     PC4     PC5     PC6
# Eigenvalue              3.9314 1.9486 0.9084 0.75157 0.57685 0.30986
# Proportion Explained    0.4368 0.2165 0.1009 0.08351 0.06409 0.03443
# Cumulative Proportion   0.4368 0.6533 0.7543 0.83777 0.90186 0.93629
#                            PC7     PC8     PC9
# Eigenvalue              0.26713 0.16252 0.14373
# Proportion Explained    0.02968 0.01806 0.01597
# Cumulative Proportion   0.96597 0.98403 1.00000
#
# Scaling 2 for species and site scores
# * Species are scaled proportional to eigenvalues
# * Sites are unscaled: weighted dispersion equal on all dimensions
```

Части результатов в `summary()`

- Importance of components — **собственные числа** (eigenvalues) и доля объясненной изменчивости
- Species scores — **факторные нагрузки исходных переменных** на каждую из компонент
- Site scores — **факторные координаты объектов**

Масштабирование — scaling

- **scaling = "species", correlation = TRUE** — отношения между переменными (нагрузки переменных пересчитаны с учетом соб. чисел, интерпретируются как корреляции)
- **scaling = "sites"** — отношения между объектами (факт. координаты пересчитаны с учетом соб. чисел)

Что нужно знать, чтобы интерпретировать результаты?

Мы хотим снизить размерность данных и вместо множества исходных признаков получить несколько главных компонент (лучше 2 или 3 для удобства интерпретации).

Эти главные компоненты будут описывать данные почти так же хорошо, как исходные признаки, но при этом будут независимы друг от друга.

Мы сможем трактовать компоненты как сложные признаки и описывать отношения между объектами в терминах этих признаков.

Чтобы все это получилось, нужно ответить на несколько вопросов:

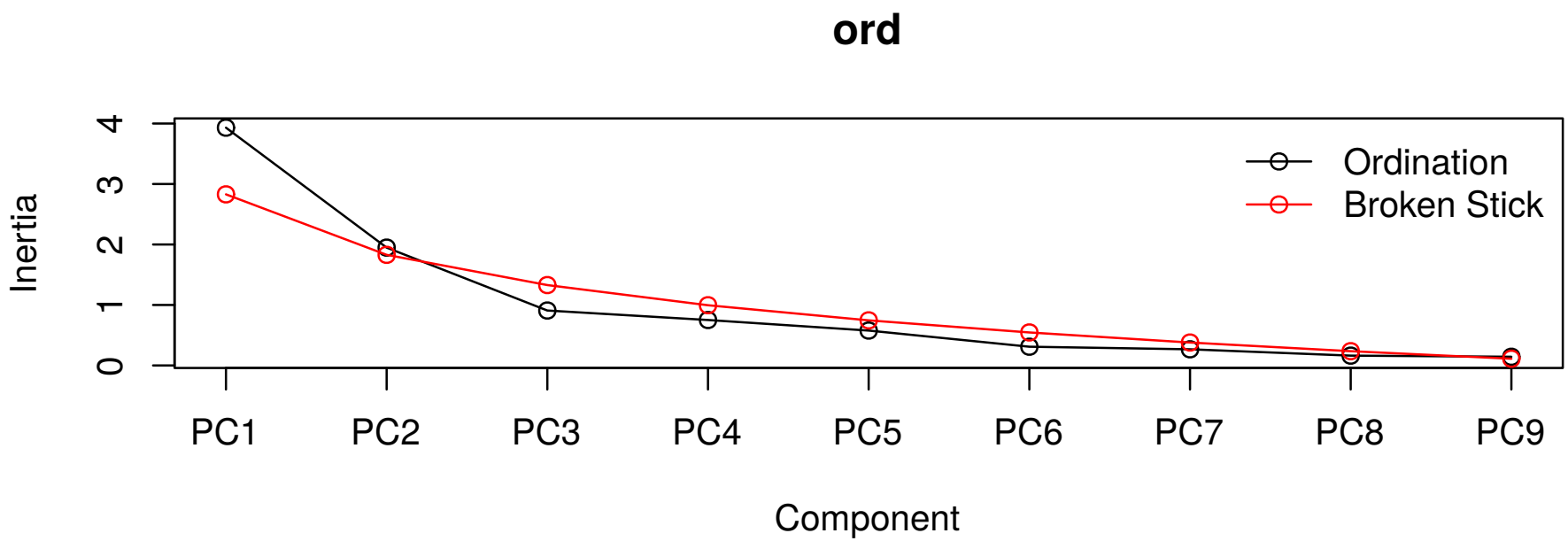
1. Сколько компонент нужно оставить?
2. Сколько общей изменчивости объясняют оставленные компоненты?
3. Что означают получившиеся компоненты?
4. Как располагаются объекты в пространстве главных компонент?

1. Сколько компонент нужно оставить?

Можно оставить только компоненты, которые объясняют больше изменчивости, чем возможно случайно (по модели сломанной палки), либо только компоненты, которые объясняют суммарно заданный процент общей изменчивости (см. далее).

Строим график собственных чисел

```
screeplot(ord, bstick = TRUE, type = 'lines')
```



2. Сколько изменчивости объясняют компоненты?

Допустим, мы решили оставить первые две компоненты.

Изменчивость, объясненная каждой из компонент, в процентах

```
eigenvals(ord) / sum(eigenvals(ord)) * 100
```

```
#      PC1      PC2      PC3      PC4      PC5      PC6      PC7
# 43.68244 21.65061 10.09291  8.35079  6.40945  3.44289  2.96814
#      PC8      PC9
#  1.80578  1.59698
```

Первые две компоненты объясняют 65 % общей изменчивости.

3. Что означают получившиеся компоненты?

Факторные нагрузки описывают связь переменных с компонентами

- Вклад переменных в изменчивость вдоль компоненты тем сильнее, чем больше модуль их факторной нагрузки.
- Знак факторной нагрузки означает направление изменения исходной переменной вдоль главной компоненты.

```
scores(ord, display = 'species', choices = c(1, 2, 3),
       scaling = 'species', correlation = TRUE)
```

```
#                 PC1         PC2          PC3
# hdlngth  -0.4713851 -0.04837773  0.078655520
# skullw   -0.4194429 -0.08480655  0.131206176
# totlngth -0.4542416 -0.05969730 -0.177801904
# taill    -0.2098116 -0.36809068 -0.279173018
# footlgth -0.3333944  0.38003868 -0.041289909
# earconch -0.1504873  0.48821273 -0.011420156
# eye      -0.2017138 -0.21130983  0.370315121
# chest    -0.4446740  0.06787162 -0.005893116
# belly    -0.3983862 -0.06276943 -0.023506174
# attr(,”const”)
# [1] 5.477226
```

3. Что означают получившиеся компоненты?

	PC1	PC2	PC3
hdlngth	**-0.4713851**	-0.04837773	0.078655520
skullw	**-0.4194429**	-0.08480655	0.131206176
totlngth	**-0.4542416**	-0.05969730	-0.177801904
taill	-0.2098116	**-0.36809068**	-0.279173018
footlgth	-0.3333944	**0.38003868**	-0.041289909
earconch	-0.1504873	**0.48821273**	-0.011420156
eye	-0.2017138	-0.21130983	**0.370315121**
chest	**-0.4446740**	0.06787162	-0.005893116
belly	**-0.3983862**	-0.06276943	-0.023506174

- PC1 — это физические размеры поссумов (высокие нагрузки у переменных длина головы, общая длина, измерения черепа, груди и живота). У нагрузок отрицательный знак, значит у крупных поссумов будут маленькие значения координат по первой компоненте.
- PC2 — длина ушей, ног и хвоста. Высокие значения по этой компоненте у поссумов с большими ушами, длинными ногами и коротким хвостом.
- PC3 — размеры глаз. Высокие значения по этой компоненте будут у поссумов с большими глазами.

Можно нарисовать факторные нагрузки на графике

- Чем ближе стрелки исходных признаков к оси компоненты, тем выше их нагрузка.
- Стрелки направлены в сторону увеличения значения исходного признака

```
biplot(ord, scaling = 'species', correlation = TRUE,
       main = 'PCA -  species scaling', display = 'species')
```

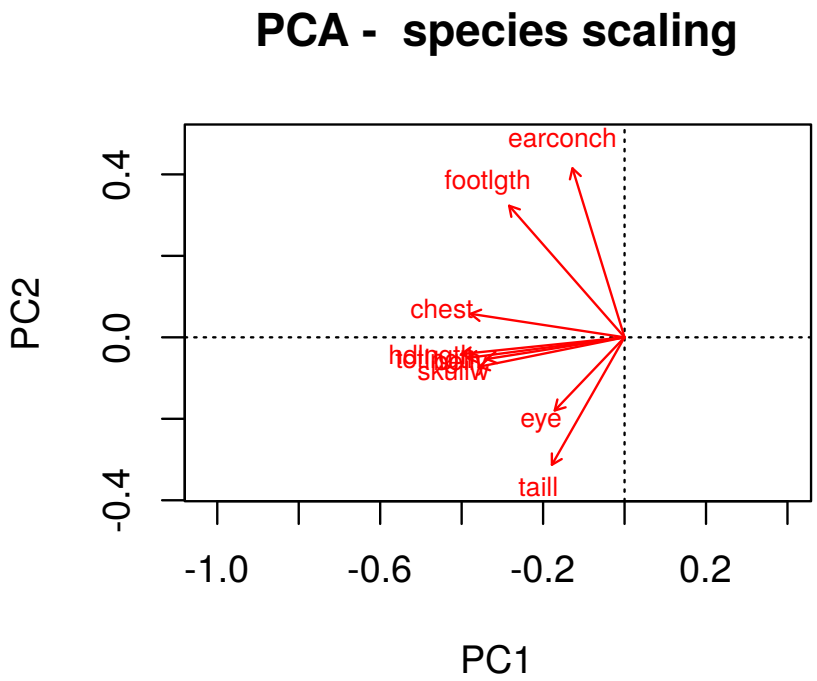
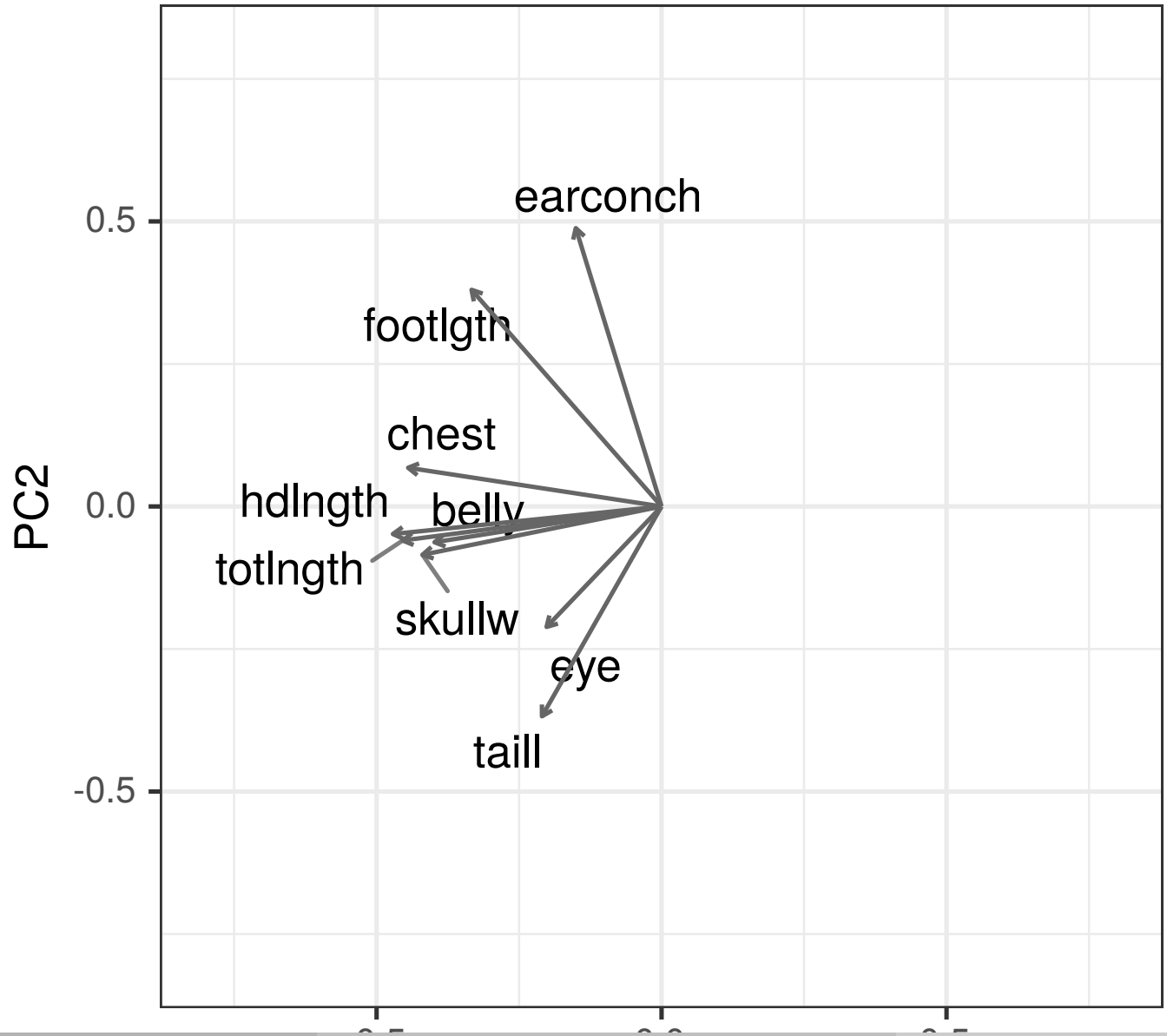


График факторных нагрузок в ggplot2

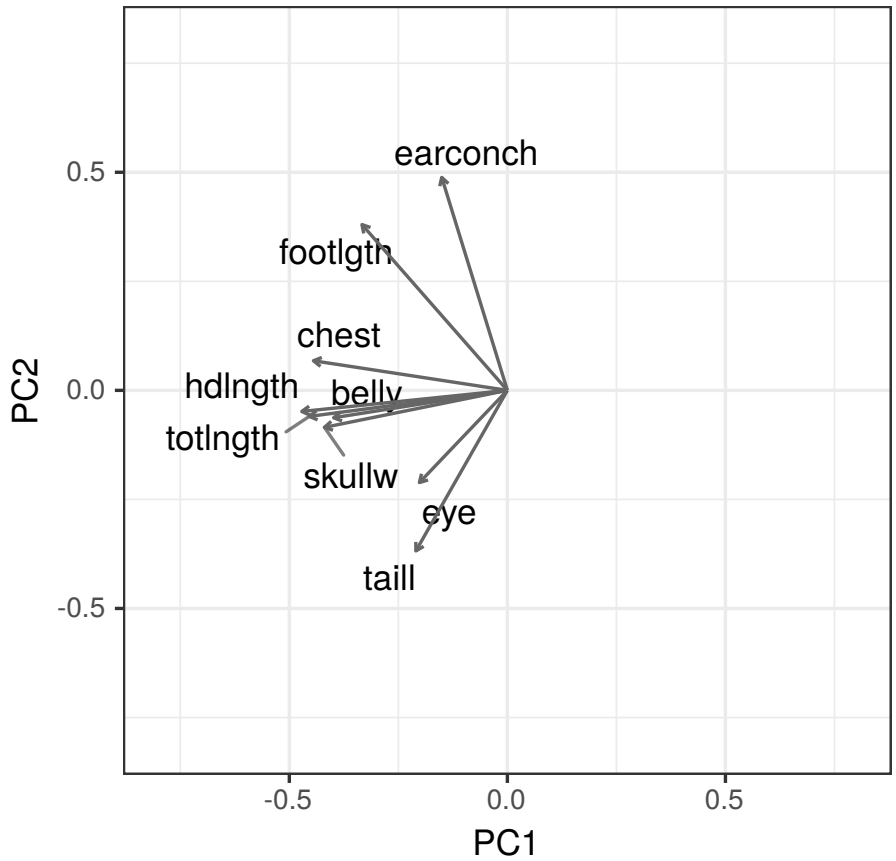
```
library(ggplot2)
theme_set(theme_bw())
library(ggrepel) # для подписей (geom_text_repel)
library(grid) # для стрелочек
# параметры стрелочек
ar <- arrow(length = unit(0.1, 'cm'))
# датафрейм с факторными нагрузками
df_load <- data.frame(scores(ord, display = 'species',
         choices = c(1, 2), scaling = 'species', correlation = TRUE))
# график
ggloadings <- ggplot(df_load) +
  geom_text_repel(aes(x = PC1, y = PC2,
    label = rownames(df_load)), segment.alpha = 0.5) +
  geom_segment(aes(x = 0, y = 0, xend = PC1, yend = PC2),
    colour = 'grey40', arrow = ar) +
  coord_equal(xlim = c(-0.8, 0.8), ylim = c(-0.8, 0.8))
ggloadings
```

График факторных нагрузок в ggplot2



Интерпретируем компоненты по графику факторных нагрузок

- PC1 — это физические размеры поссумов (высокие нагрузки у переменных длина головы, общая длина, измерения черепа, груди и живота). У нагрузок отрицательный знак, значит у крупных поссумов будут маленькие значения координат по первой компоненте.
- PC2 — длина ушей, ног и хвоста. Высокие значения по этой компоненте у поссумов с большими ушами, длинными ногами и коротким хвостом.



4. Значения факторов (= факторные координаты) — координаты объектов в пространстве главных компонент

```
# Координаты можно добыть так (но сейчас нам нужен только график)
scores(ord, display = 'sites',  choices = c(1, 2, 3), scaling = 'sites')
```

```
#                PC1          PC2           PC3
# C3    -0.360819375  0.304131424  0.0688821891
# C5    -0.240088432  0.143367378  0.0693525874
# C10   -0.551140377  0.107837459 -0.1423865805
# C15   -0.312799927  0.207126158 -0.1273986533
# C23   -0.034485559  0.267285182  0.0488052857
# C24   -0.151004588  0.394274329 -0.1264112959
# C26   -0.298396559  0.251333661 -0.0692674630
# C27   -0.306099050  0.266871815 -0.1203631232
# C28   -0.214566258  0.193184218 -0.0296039420
# C31   -0.095501125  0.209273041 -0.1387304801
# C32   -0.399460404  0.205195000 -0.1800913882
# C34   -0.187568929  0.198586258  0.0183336106
# C36   -0.191356099  0.080859321  0.1644518057
# C37   -0.250735386  0.312058901  0.0033603952
# C39   -0.084377646  0.232837138  0.2515360165
# C40    0.039106450  0.358378412  0.0251344458
# C45   -0.431885643  0.225796864  0.1077821334
# C47   -0.159125483  0.359986504  0.0058833057
```

График факторных координат (= график ординации)

```
biplot(ord, scaling = 'sites', display = 'sites',
       type = 't', main = 'PCA - sites scaling')
```

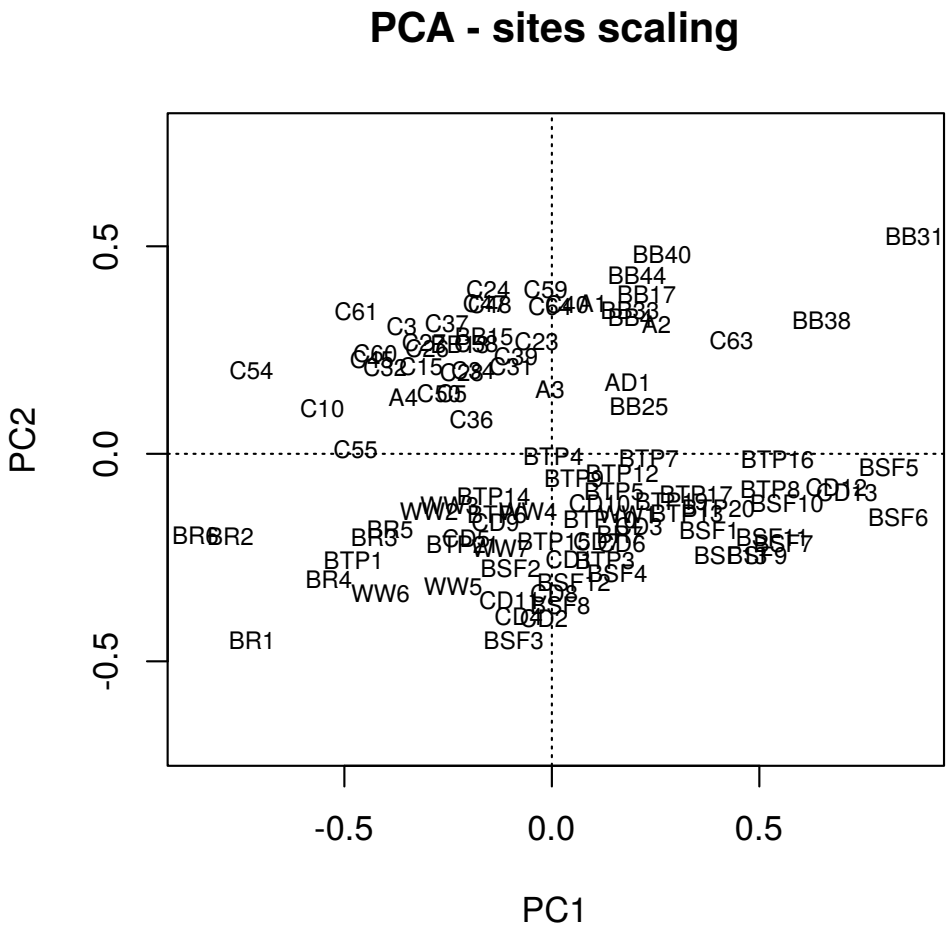
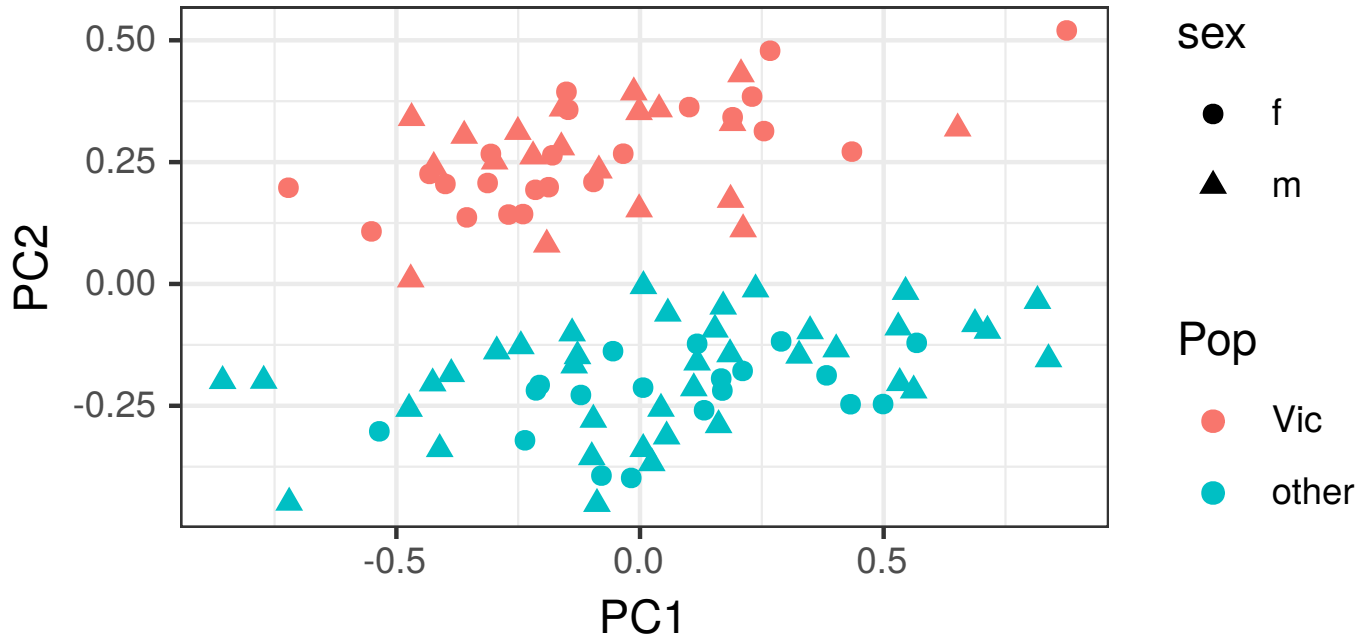


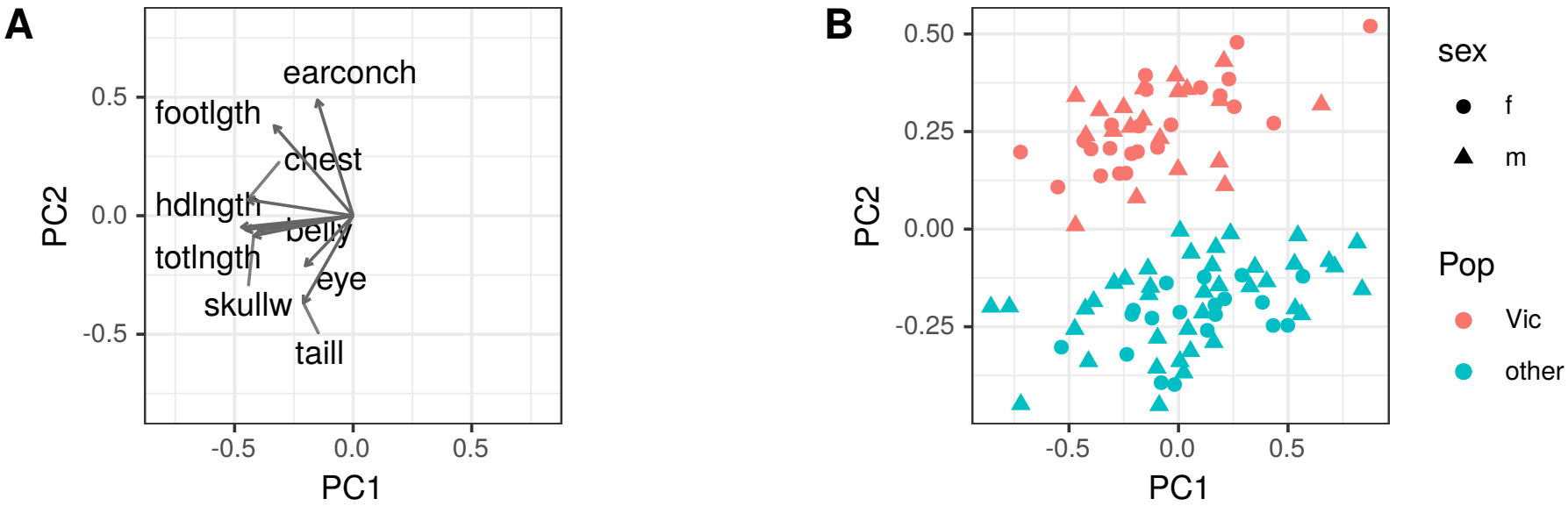
График факторных координат в ggplot2

```
# данные для графика: факторные координаты и исходные переменные
df_scores <- data.frame(pos,
                        scores(ord, display = 'sites', scaling = 'sites',
                               choices = c(1, 2, 3)))
# график ординации
ggscores <- ggplot(df_scores, aes(x = PC1, y = PC2,
                                  colour = Pop, shape = sex)) +
  geom_point(size = 2) + coord_equal()
ggscores
```



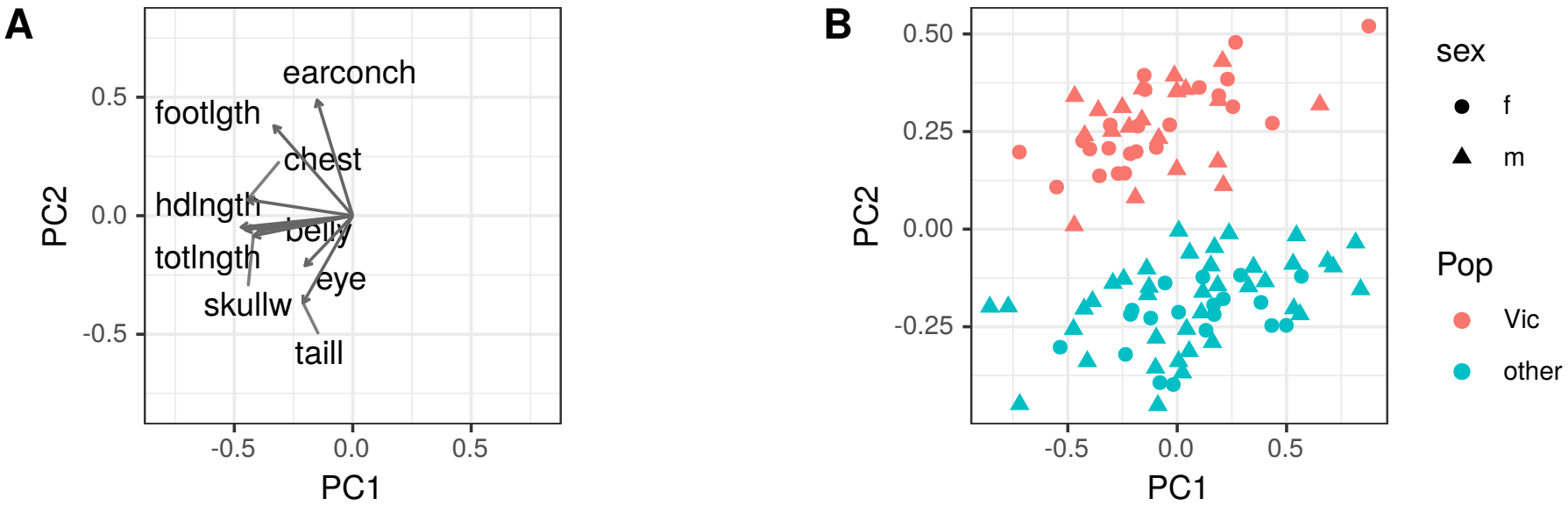
Для удобства интерпретации ординации, располагаем ее рядом с графиком факторных нагрузок

```
library(cowplot)
plot_grid(ggloadings, ggscores, labels = 'AUTO', align = 'hv', axis = 'r')
```



Для удобства интерпретации ординации, располагаем ее рядом с графиком факторных нагрузок

```
library(cowplot)
plot_grid(ggloadings, ggscores, labels = 'AUTO', align = 'hv', axis = 'r')
```



Первые две компоненты объясняют 65% общей изменчивости. Первая компонента (44%) связана с размером особей. Вторая компонента (21%) описывает пропорции ног, ушей и хвоста. Внутри популяций поссумы мало различаются по этим параметрам (об этом говорит небольшой разброс точек вдоль второй компоненты). Зато поссумы из провинции Виктория не похожи на поссумов из других провинций: у них относительно более крупные уши, длинные ноги и короткие хвосты.

Факторные координаты можно использовать для снижения размерности данных

Было 7 скоррелированных признаков, стало 2 **независимых** (они ведь перпендикулярны) главных компоненты

Значения факторных координат можно использовать в анализах, где нужна независимость переменных:

- Множественная регрессия
- Дискриминантный анализ (например, генетические данные)
- Дисперсионный анализ
- Корреляция с другими признаками, которые не были использованы в анализе главных компонент, и т.д., и т.п.

Условия применимости анализа главных компонент

Похожи на условия применимости множественной линейной регрессии

- Линейные связи между переменными (т.к. матрица корреляций или ковариаций)
- Исключить наблюдения, в которых есть пропущенные значения
- Если много нулей — трансформация данных (например, трансформация Хелингера)
- Если очень много нулей — удалить такие переменные из анализа

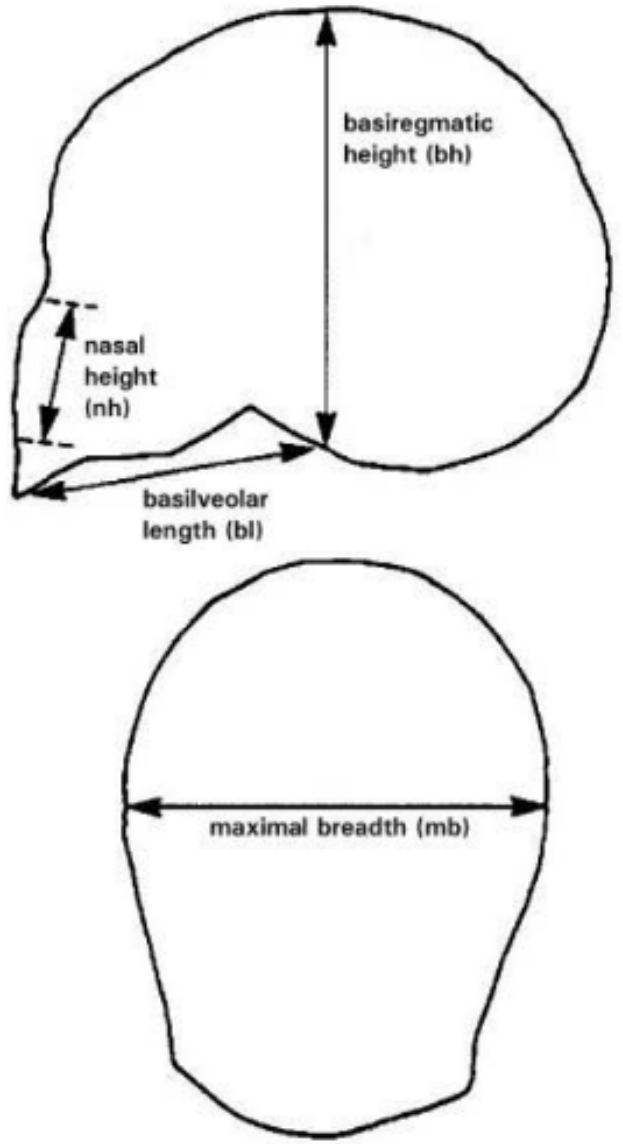
Пример: Морфометрия египетских черепов

Измерения 150 черепов в мм:

- mb — максимальная ширина
- bh — высота от основания до макушки
- bl — расстояние от основания черепа до края в. челюсти
- nh — высота носа

Эпоха (epoch):

- 1 — ранний прединастический период (ок. 4000 до н.э.)
- 2 — поздний прединастический период (ок. 3300 до н.э.)
- 3 — 12 и 13 династии (ок. 1850 до н.э.)
- 4 — Птолемейский период (ок. 200 до н.э.)
- 5 — Римский период (ок. 150 н.э.)



{Данные Thompson, Randall-Maciver (1905). Источник Manly (1994).}

Знакомимся с данными

```
library(HSAUR)
data('skulls')
str(skulls)
```

```
# 'data.frame': 150 obs. of  5 variables:
# $ epoch: Ord.factor w/ 5 levels "c4000BC"<"c3300BC"<..: 1 1 1 1 1 1 1 1 1 1 ...
#  $ mb   : num  131 125 131 119 136 138 139 125 131 134 ...
#  $ bh   : num  138 131 132 132 143 137 130 136 134 134 ...
#  $ bl   : num  89 92 99 96 100 89 108 93 102 99 ...
#  $ nh   : num  49 48 50 44 54 56 48 48 51 51 ...
```

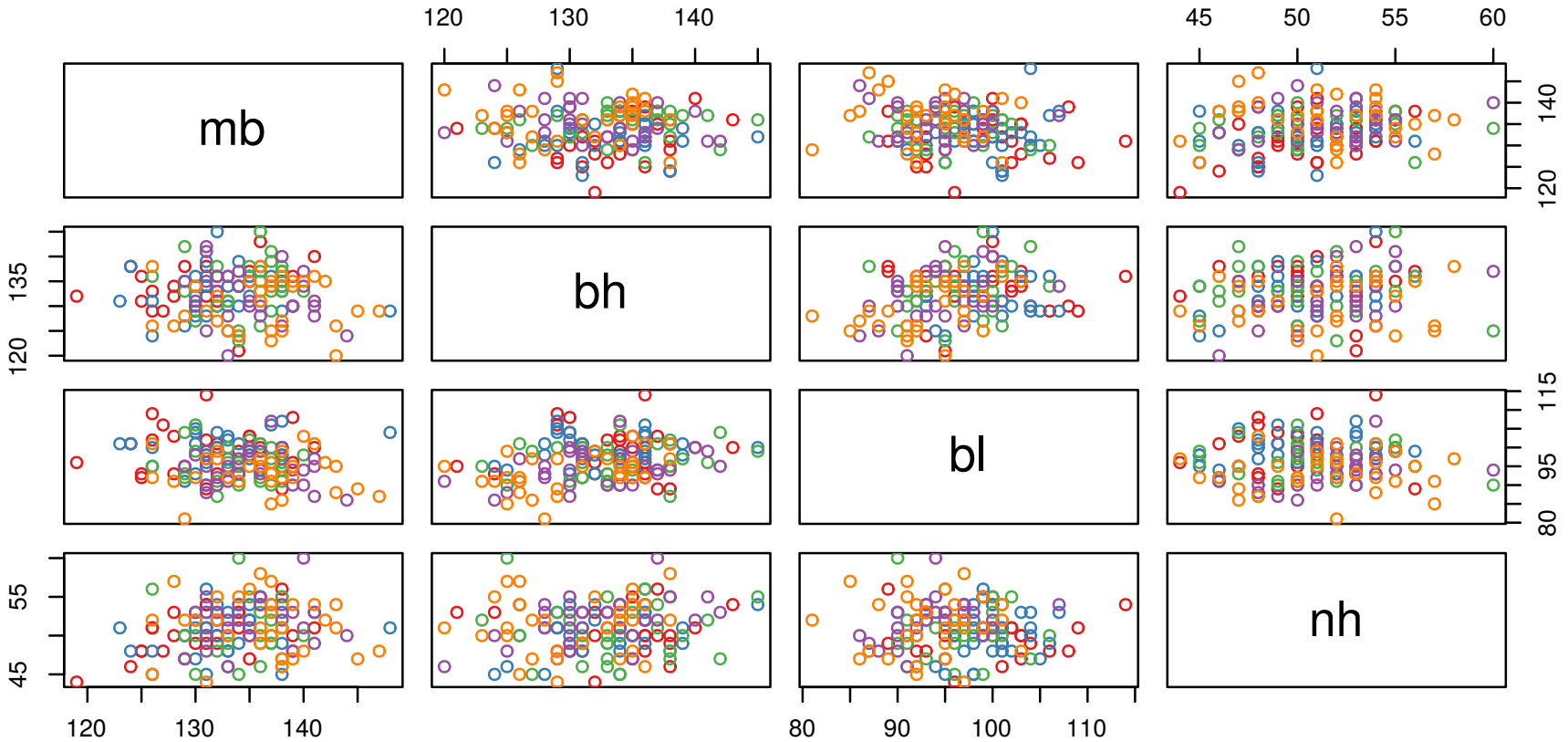
```
sum(is.na(skulls))
```

```
# [1] 0
```

```
table(skulls$epoch)
```

```
#
# c4000BC c3300BC c1850BC  c200BC  cAD150
#      30      30      30      30      30
```

```
# цвета
library(RColorBrewer)
cols <- brewer.pal(n = length(levels(skulls$epoch)), name = 'Set1')
# график
pairs(skulls[, -1], col = cols[skulls$epoch])
```



Задание 1

Сделайте анализ главных компонент.

1. Сколько компонент нужно оставить?
2. Сколько общей изменчивости объясняют оставленные компоненты?
3. Что означают получившиеся компоненты?
4. Как располагаются объекты в пространстве главных компонент?

Как менялась форма черепов в древнем египте в разные эпохи?

Решение

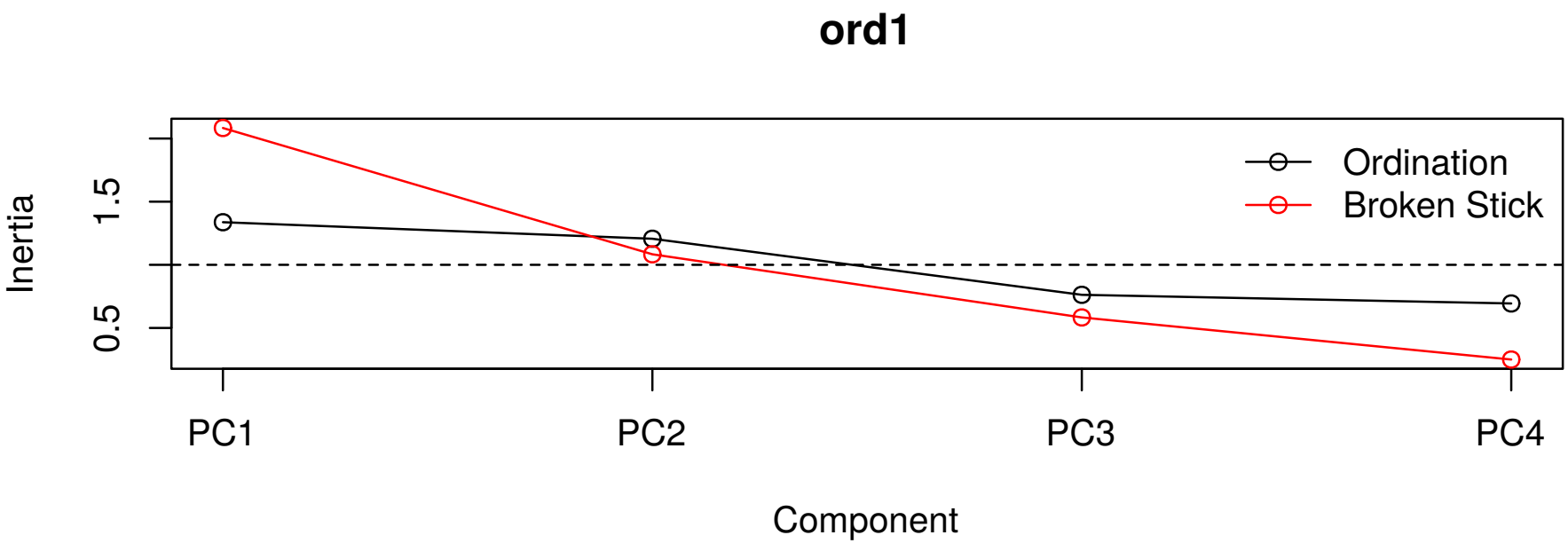
Делаем анализ главных компонент.

Не забудьте оставить в исходных данных только непрерывные переменные

```
ord1 <- rda(skulls[, -1], scale = TRUE)
```

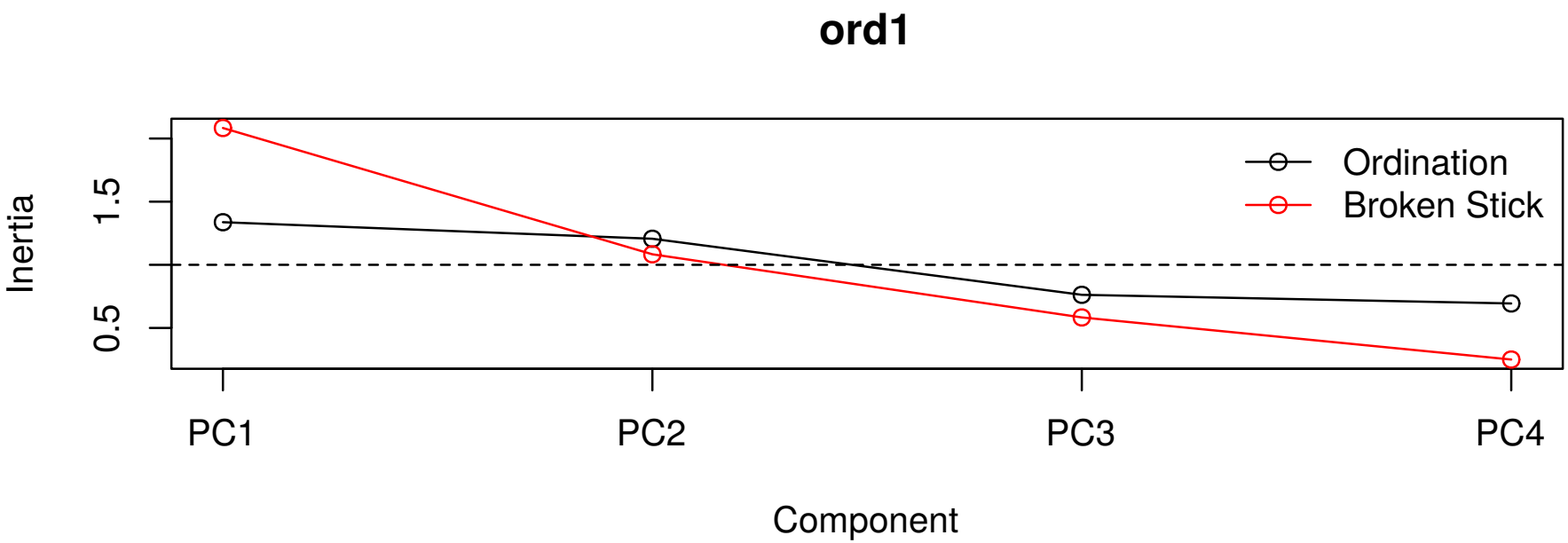
Сколько компонент нужно оставить?

```
screeplot(ord1, bstick = TRUE, type = 'lines')
abline(h = 1, lty = 2)
```



Сколько компонент нужно оставить?

```
screeplot(ord1, bstick = TRUE, type = 'lines')
abline(h = 1, lty = 2)
```



- Оставляем две компоненты (можно даже одну, но это будет сложно нарисовать)

Сколько изменчивости объясняют компоненты?

```
eig <- eigenvals(ord1)
explained <- sum(eig[1:2])/sum(eig) * 100
explained

# [1] 63.59221
```

Сколько изменчивости объясняют компоненты?

```
eig <- eigenvals(ord1)
explained <- sum(eig[1:2])/sum(eig) * 100
explained
```

```
# [1] 63.59221
```

- Компоненты вместе объясняют 64 % общей изменчивости

Что означают получившиеся компоненты?

- Вдоль 1й компоненты уменьшается расстояние от основания черепа до края в. челюсти (bl) и высота от основания до макушки (bh)
- Вдоль 2й компоненты уменьшается высота носа (nh) и максимальная ширина (mb)

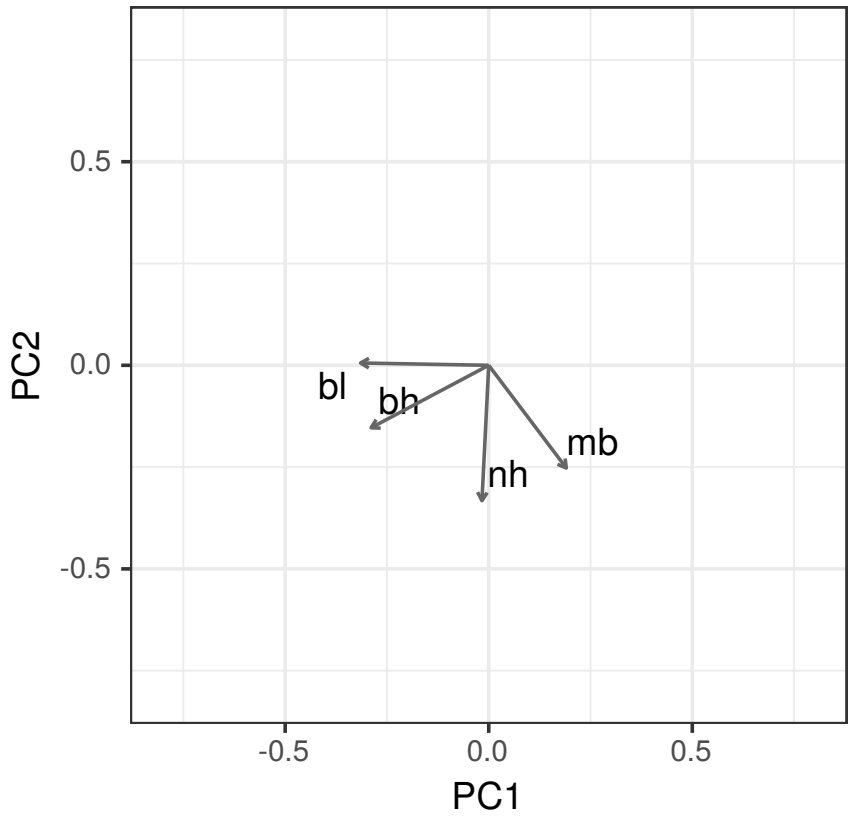
```
scores(ord1, display = 'species', choices = c(1, 2),
       scaling = 'species', correlation = TRUE)
```

```
#             PC1          PC2
# mb  0.19050654 -0.252267512
# bh -0.28889919 -0.153390201
# bl -0.31476765  0.005671088
# nh -0.01663669 -0.332360682
# attr(,”const”)
# [1] 4.940963
```

Что означают получившиеся компоненты?

```
# данные для графика факторных нагрузок
df_load <- data.frame(scores(ord1, display = 'species',
                             choices = c(1, 2, 3),
                             scaling = 'species', correlation = TRUE))
# стрелочки
ar <- arrow(length = unit(0.1, 'cm'))
# график факторных нагрузок
ggloadings1 <- ggplot(df_load) +
  geom_text_repel(aes(x = PC1, y = PC2,
                      label = rownames(df_load))) +
  geom_segment(aes(x = 0, y = 0, xend = PC1, yend = PC2),
               colour = 'grey40', arrow = ar) +
  coord_equal(xlim = c(-0.8, 0.8), ylim = c(-0.8, 0.8))
ggloadings1
```

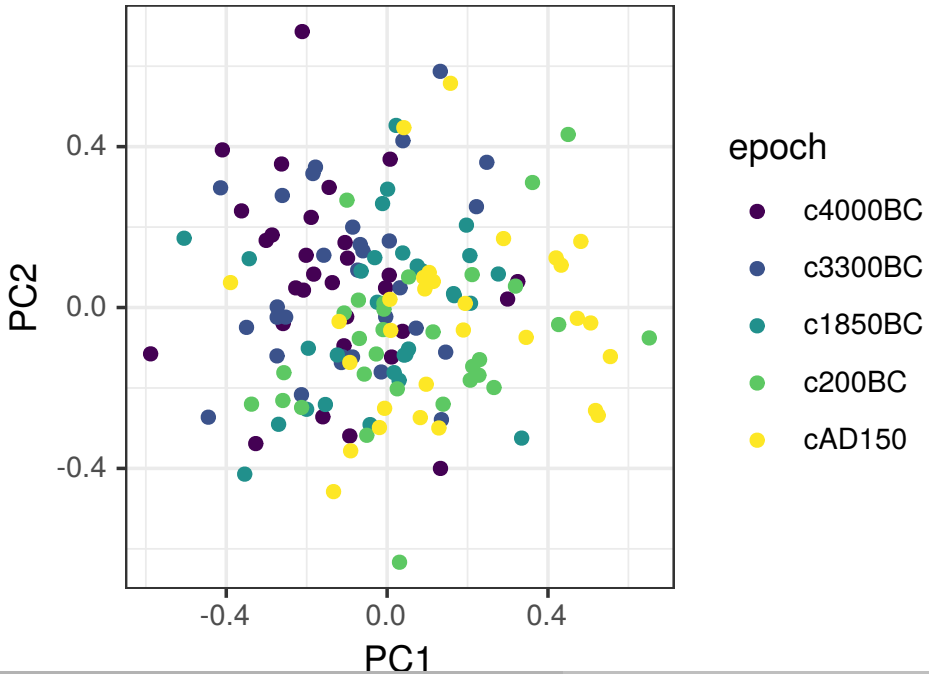
Что означают получившиеся компоненты?



- Вдоль 1й компоненты уменьшается расстояние от основания черепа до края в. челюсти (bl) и высота от основания до макушки (bh)
- Вдоль 2й компоненты уменьшается высота носа (nh) и максимальная ширина (mb)

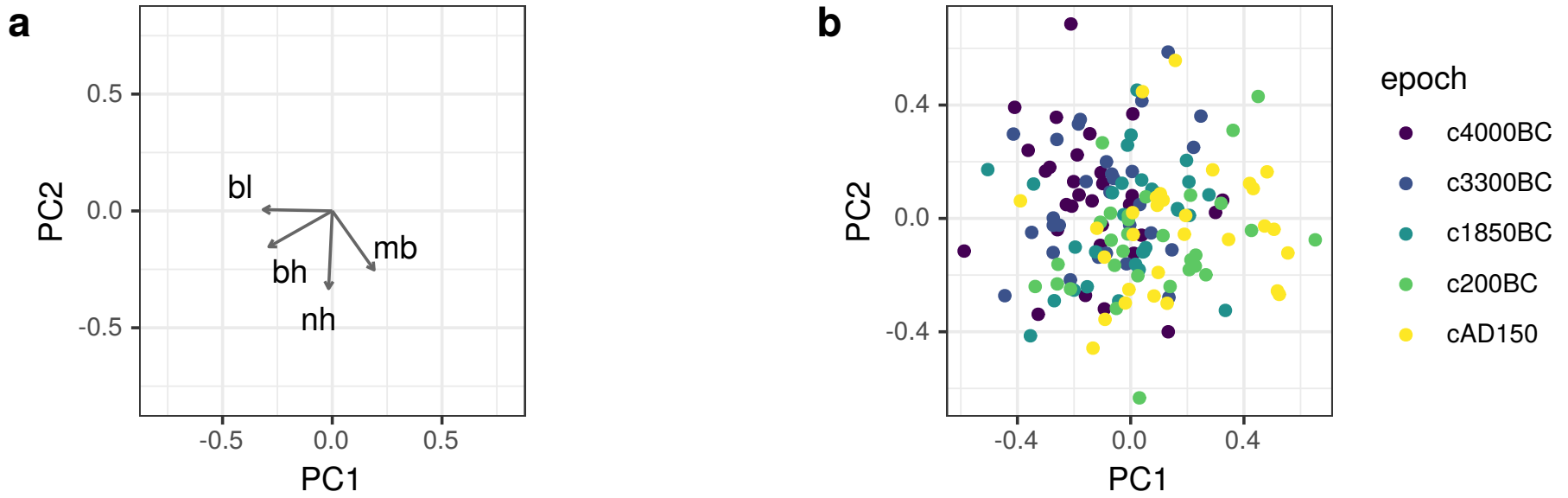
Как располагаются объекты в пространстве главных компонент?

```
# Данные для графика ординации
df_scores1 <- data.frame(skulls,
                         scores(ord1, display = 'sites',
                                choices = c(1, 2), scaling = 'sites'))
# График ординации
ggscores1 <- ggplot(df_scores1, aes(x = PC1, y = PC2)) +
  geom_point(aes(colour = epoch)) + coord_equal()
ggscores1
```



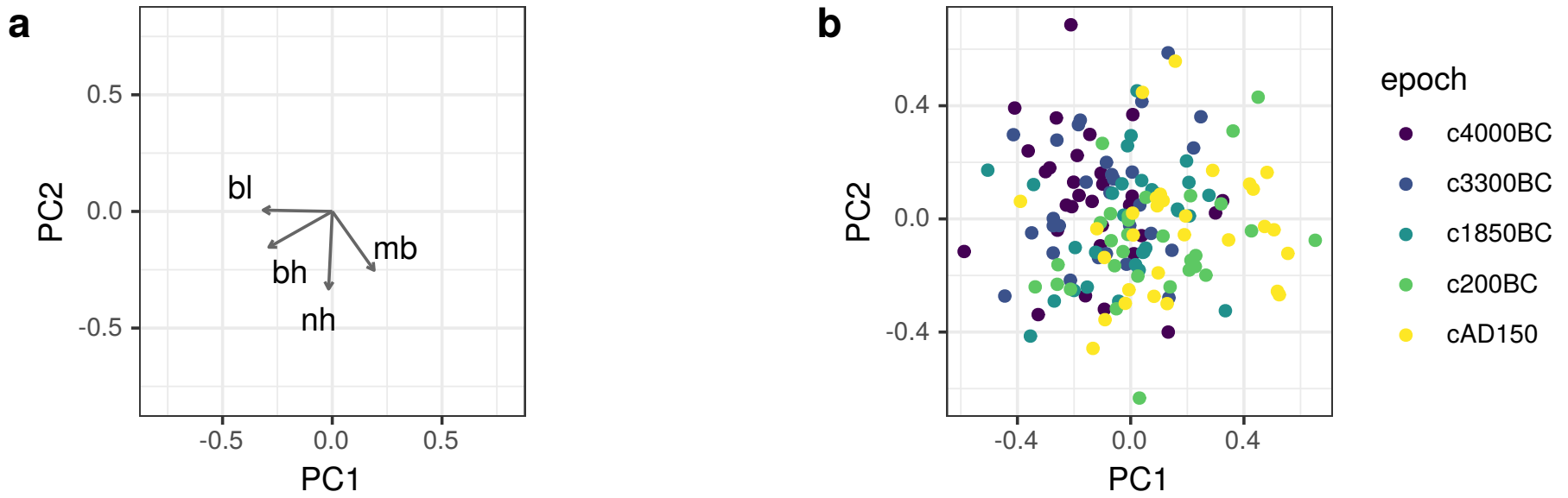
Для удобства интерпретации ординации, располагаем ее рядом с графиком факторных нагрузок

```
# library(cowplot)
plot_grid(ggloadings1, ggscores1, labels = 'auto', align = 'vh', axis = 'r')
```



Для удобства интерпретации ординации, располагаем ее рядом с графиком факторных нагрузок

```
# library(cowplot)
plot_grid(ggloadings1, ggscores1, labels = 'auto', align = 'vh', axis = 'r')
```



- С течением времени форма черепов древних египтян менялась. Размеры черепа постепенно увеличивались, а длина носа практически не изменялась.

Take-home messages

- Метод главных компонент:
  - исследование связей между переменными
  - построение ординации объектов
  - снижение размерности данных
- Собственные числа — вклад компонент в общую изменчивость
- Факторные нагрузки — связь исходных переменных с компонентами — используются для интерпретации
- Значения факторов (факторные координаты) - новые координаты объектов в пространстве уменьшенной размерности
- Значения факторов можно использовать как новые комплексные переменные в других видах анализов.

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.
- The Ordination Web Page URL http://ordination.okstate.edu/ (accessed 10.21.13).
- Quinn, G.G.P., Keough, M.J., 2002. Experimental design and data analysis for biologists. Cambridge University Press.
- Zuur, A.F., Ieno, E.N., Smith, G.M., 2007. Analysing ecological data. Springer.