# Линейные модели с дискретными предикторами

## Линейные модели...

Марина Варфоломеева, Вадим Хайтов

Кафедра Зоологии беспозвоночных, Биологический факультет, СПбГУ

# Линейные модели с дискретными предикторами (дисперсионный анализ)

## Вы сможете

- Объяснить, в чем опасность множественных сравнений, и как с ними можно бороться
- Рассказать, как в дисперсионном анализе моделируются значения зависимой переменной
- Интерпретировать и описать результаты, записанные в таблице дисперсионного анализа
- Перечислить и проверить условия применимости дисперсионного анализа
- Провести множественные попарные сравнения при помощи post hoc теста Тьюки, представить и описать их результаты
- Построить график результатов дисперсионного анализа

# Дисперсионный анализ (Analysis Of Variance, ANOVA)

**Дисперсионный анализ в широком смысле** — анализ изменений непрерывной зависимой переменной в связи с разными источниками изменчивости (предикторами).

Мы использовали его для тестирования значимости предикторов в линейных моделях.

**Дисперсионный анализ в узком смысле** — это частный случай, когда в линейной модели используются только дискретные предикторы (факторы).

Он используется для сравнения средних значений зависимой переменной в дискретных группах, заданных факторами..

# Пример: яйца кукушек

Различаются ли размеры яиц кукушек в гнездах разных птиц-хозяев?

Датасет `cuckoos` из пакета `DAAG`:

- `species` — вид птиц-хозяев (фактор)
- `length` — длина яиц кукушек в гнездах хозяев (зависимая переменная)

Данные: Latter, 1902; источник: Tippett, 1931

# Открываем данные

```
library(DAAG)
data("cuckoos")
# Положим данные в переменную с коротким названием, чтобы меньше печатать
cu <- cuckoos
head(cu, 3)
```

```
#   length breadth      species id
# 1   21.7    16.1 meadow.pipit 21
# 2   22.6    17.0 meadow.pipit 22
# 3   20.9    16.2 meadow.pipit 23
```

```
# Сократим названия переменных
colnames(cu) <- c('len', 'br', 'sp', 'id')
```

# Изменим названия уровней фактора, чтобы было легче понять о каких птицах речь

```
levels(cu$sp)
```

```
# [1] "hedge.sparrow" "meadow.pipit"  "pied.wagtail"  "robin"
# [5] "tree.pipit"    "wren"
```

```
levels(cu$sp) <- c("ЛесЗав", "ЛугКон", "БелТряс",
                   "Малин", "ЛесКон", "Крапив")
```

# Исследуем данные

```r
# Пропущенных значений нет
colSums(is.na(cu))
```

```
# len  br  sp  id
#   0   0   0   0
```

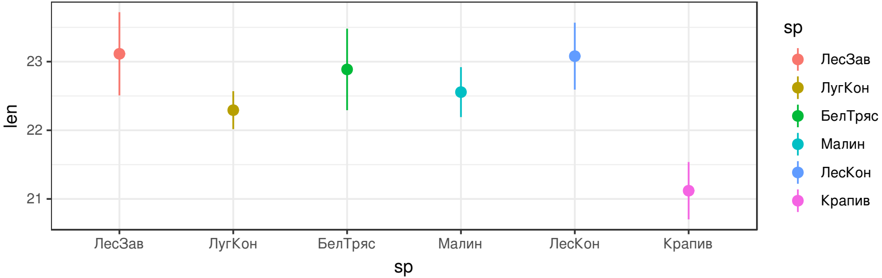
```r
# Данные не сбалансированы (размеры групп разные)
table(cu$sp)
```

```
#
#  ЛесЗав  ЛугКон БелТряс   Малин  ЛесКон  Крапив
#      14      45      15      16      15      15
```

# Задание

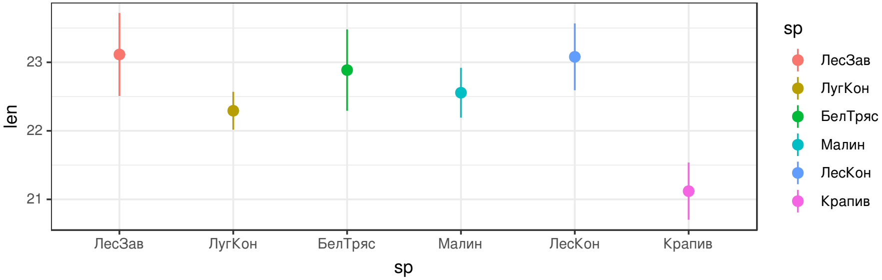
Дополните код, чтобы построить график зависимости размера яиц кукушек (`len`) от вида птиц-хозяев (`sp`), в гнездах которых были обнаружены яйца. На графике должны быть изображены средние значения и их 95% доверительные интервалы, а цвет должен соответствовать виду птиц-хозяев.

```
theme_set( )
ggplot(data = , aes()) +
  stat_summary(geom = "pointrange", fun.data = mean_cl_normal)
```



# Решение

```
library(ggplot2)
theme_set(theme_bw())
ggplot(data = cu, aes(x = sp, y = len, colour = sp)) +
  stat_summary(geom = "pointrange", fun.data = mean_cl_normal)
```



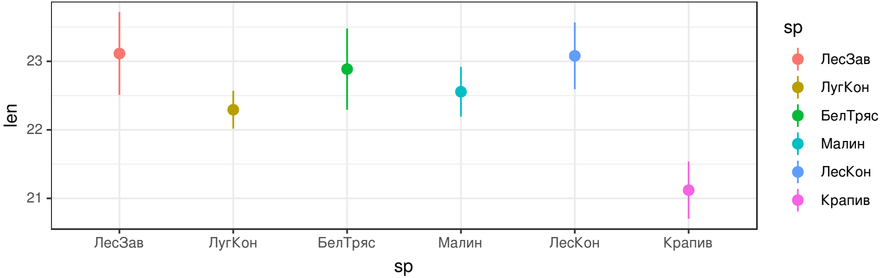
# "Некрасивый" порядок уровней на графике

На этом графике некрасивый порядок уровней: средние для разных уровней фактора cu$sp расположены, как кажется, хаотично.

Порядок групп на графике определяется порядком уровней фактора.

```
# "старый" порядок уровней
levels(cu$sp)
```

```
# [1] "ЛесЗав"  "ЛугКон"  "БелТряс" "Малин"   "ЛесКон"  "Крапив"
```



# Меняем порядок уровней

Давайте изменим порядок уровней в факторе `cu$sp` так, чтобы он соответствовал возрастанию средних значений длины яиц `cu$len`.

```
# "старый" порядок уровней
levels(cu$sp)
```

```
# [1] "ЛесЗав"  "ЛугКон"  "БелТряс" "Малин"   "ЛесКон"  "Крапив"
```

```
# переставляем уровни в порядке следования средних значений
cu$sp <- reorder(cu$sp, cu$len, FUN = mean)
# "новый" порядок уровней стал таким
levels(cu$sp)
```

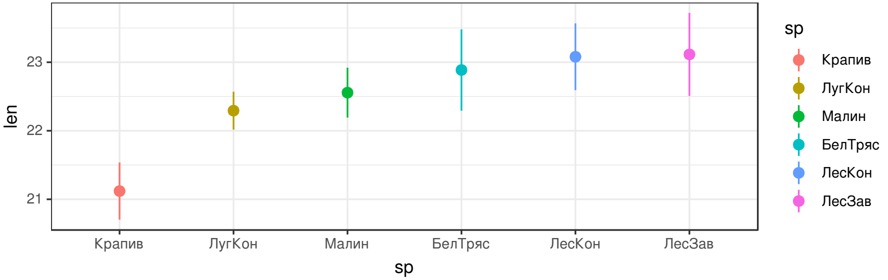
```
# [1] "Крапив"  "ЛугКон"  "Малин"   "БелТряс" "ЛесКон"  "ЛесЗав"
```

# График с новым порядком уровней

С новым порядком уровней нам легче визуально сравнивать друг с другом категории.

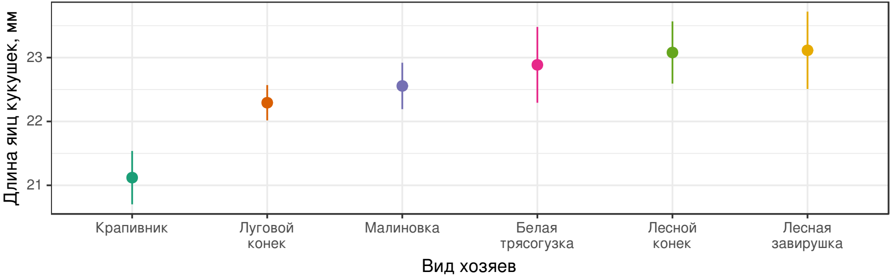
Поскольку, изменив порядок уровней, мы внесли изменения в исходные данные, придется полностью обновить график (т.к.`ggplot()` хранит данные внутри графика).

```
ggplot(data = cu, aes(x = sp, y = len, colour = sp)) +
  stat_summary(geom = "pointrange", fun.data = mean_cl_normal)
```



# Понравившийся график, если понадобится, можно в любой момент довести до ума, а остальные удалить

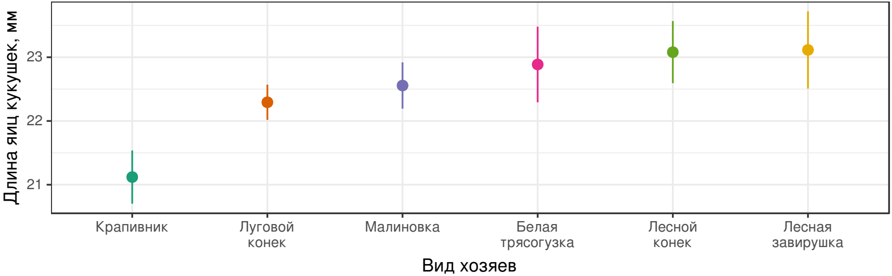
```
ggplot(data = cu, aes(x = sp, y = len, colour = sp)) +
  stat_summary(geom = "pointrange", fun.data = mean_cl_normal) +
  labs(x = "Вид хозяев", y = "Длина яиц кукушек, мм") +
  scale_colour_brewer(name = "Вид \nхозяев", palette = "Dark2") +
  scale_x_discrete(labels = c("Крапивник", "Луговой\nконек", "Малиновка",
"Белая\nтрясогузка", "Лесной\nконек", "Лесная\nзавирушка")) +
  theme(legend.position = "none")
```



# Множественные сравнения

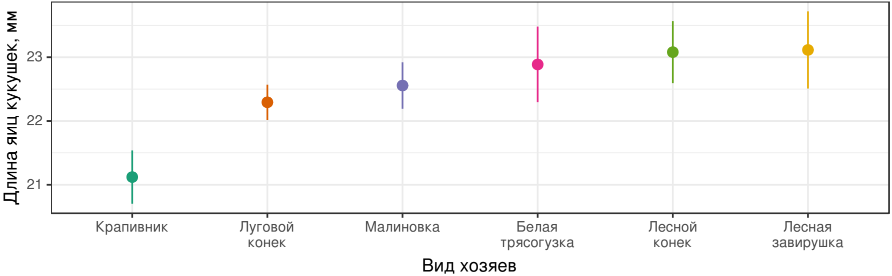
# Множественные сравнения

Мы могли бы сравнить длину яиц в гнездах резных хозяев при помощи t-критерия. У нас всего 6 групп. Сколько возможно между ними попарных сравнений?



# Множественные сравнения

Мы могли бы сравнить длину яиц в гнездах резных хозяев при помощи t-критерия. У нас всего 6 групп. Сколько возможно между ними попарных сравнений?

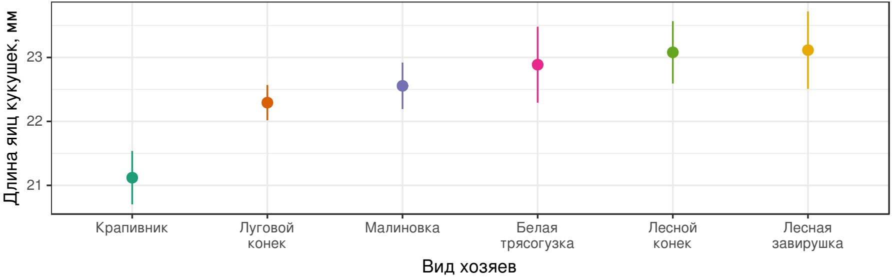



Всего возможно 15 сравнений.

Если для каждого сравнения вероятность ошибки первого рода будет $\alpha_{per\ comparison} = 0.05$, то для группы из 15 сравнений — ?

# Множественные сравнения

Мы могли бы сравнить длину яиц в гнездах резных хозяев при помощи t-критерия. У нас всего 6 групп. Сколько возможно между ними попарных сравнений?




Всего возможно 15 сравнений.

Если для каждого сравнения вероятность ошибки первого рода будет $\alpha_{per\ comparison} = 0.05$, то для группы из 15 сравнений — ?

Если предположить, что сравнения независимы (это не так), то $\alpha_{family\ wise} = 1 - (1 - 0.05)^{15} = 0.54$. Мы рискуем найти различия там где их нет с 54% вероятностью!

Для зависимых сравнений вероятность будет немного меньше, но все равно значительно больше $0.05$

# Поправка Бонферрони — очень жесткий способ коррекции.

Если нужно много сравнений, можно снизить $\alpha_{per\ comparison}$ до общепринятого уровня

$$\alpha_{per\ comparison} = \frac{\alpha_{family\ wise}}{n}$$

# Поправка Бонферрони — очень жесткий способ коррекции.

Если нужно много сравнений, можно снизить $\alpha_{per\ comparison}$ до общепринятого уровня

$$\alpha_{per\ comparison} = \frac{\alpha_{family\ wise}}{n}$$

Например, если хотим зафиксировать $\alpha_{family\ wise} = 0.05$

С поправкой Бонферрони $\alpha_{per\ comparison} = 0.05/15 = 0.003$

Это очень жесткая поправка! Мы рискуем не найти достоверных различий, даже там, где они есть…

Но есть выход. Вместо множества попарных сравнений можно использовать один тест — дисперсионный анализ (analysis of variation, ANOVA).

# Линейные модели с дискретными предикторами

# Для кодирования дискретных факторов в R используются две параметризации

**Параметризация индикаторных переменных** (dummy coding, treatment parametrization, reference cell model) в R обозначается **contr.treatment**.
С ней вы уже знакомы. Используется по умолчанию в R.

**Параметризация эффектов** (effects coding, sum-to-zero parameterization) в R обозначается **contr.sum**.
“Классическая” параметризация для дисперсионного анализа. Нужна, если хочется использовать т.наз. III тип сумм квадратов в многофакторном дисперсионном анализе со взаимодействием факторов.

# Параметризация индикаторных переменных

# Переменные-индикаторы

| Фактор | Переменные-индикаторы | | | | |
|---|---|---|---|---|---|
| sp | spЛугКон $x_1$ | spМалин $x_2$ | spБелТряс $x_3$ | spЛесКон $x_4$ | spЛесЗав $x_5$ |
| Крапив | 0 | 0 | 0 | 0 | 0 |
| ЛугКон | 1 | 0 | 0 | 0 | 0 |
| Малин | 0 | 1 | 0 | 0 | 0 |
| БелТряс | 0 | 0 | 1 | 0 | 0 |
| ЛесКон | 0 | 0 | 0 | 1 | 0 |
| ЛесЗав | 0 | 0 | 0 | 0 | 1 |

Переменных-индикаторов всегда на одну меньше, чем число уровней фактора.

Уровень “Крапив” будет базовым: для его кодирования не нужна отдельная переменная.

# Уравнение модели в параметризации индикаторов

| **Фактор** | **Переменные-индикаторы** | | | | |
|---|---|---|---|---|---|
| `sp` | `spЛугКон` $x_1$ | `spМалин` $x_2$ | `spБелТряс` $x_3$ | `spЛесКон` $x_4$ | `spЛесЗав` $x_5$ |
| Крапив | 0 | 0 | 0 | 0 | 0 |
| ЛугКон | 1 | 0 | 0 | 0 | 0 |
| Малин | 0 | 1 | 0 | 0 | 0 |
| БелТряс | 0 | 0 | 1 | 0 | 0 |
| ЛесКон | 0 | 0 | 0 | 1 | 0 |
| ЛесЗав | 0 | 0 | 0 | 0 | 1 |

$$y_i = b_0 + b_1 x_{1i} + ... + b_5 x_{5i} + e_i$$

- $b_0$ — это среднее значение отклика для базового уровня фактора.
- $b_1, ..., b_5$ — это отклонения от базового уровня для средних с другими уровнями фактора.

# Подбираем коэффициенты модели в параметризации индикаторов

```
mod_treatment <- lm(len ~ sp, data = cu)
coef(mod_treatment)
```

```
# (Intercept)    spЛугКон     spМалин   spБелТряс    spЛесКон    spЛесЗав
#   21.120000    1.173333    1.436250    1.766667    1.960000    1.994286
```

# Уравнение модели в параметризации индикаторов

```
coef(mod_treatment)
```

```
# (Intercept)    spЛугКон     spМалин    spБелТряс     spЛесКон     spЛесЗав
#   21.120000    1.173333    1.436250     1.766667    1.960000     1.994286
```

$$\widehat{len}_i = 21.12 + 1.17 sp_{\text{ЛугКон } i} + 1.44 sp_{\text{Малин } i} + 1.77 sp_{\text{БелТряс } i} + 1.96 sp_{\text{ЛесКон } i} + 1.99 sp_{\text{ЛесЗав } i}$$

Первый коэффициент — средний размер яиц кукушек в гнездах крапивников (на базовом уровне):

- $\widehat{len}_{\text{Крапив } i} = 21.12$

Другие коэффициенты — разница размеров яиц кукушек в гнездах других хозяев и в гнездах крапивников (отклонения от базового уровня):

- $\widehat{len}_{\text{ЛугКон } i} = 21.12 + 1.17 sp_{\text{ЛугКон } i} = 22.29$
- $\widehat{len}_i = 21.12 + 1.44 sp_{\text{Малин } i} = 22.56$
- $\widehat{len}_i = 21.12 + 1.77 sp_{\text{БелТряс } i} = 22.89$
- $\widehat{len}_i = 21.12 + 1.96 sp_{\text{ЛесКон } i} = 23.08$
- $\widehat{len}_i = 21.12 + 1.99 sp_{\text{ЛесЗав } i} = 23.11$

# Параметризация эффектов

# Переменные-эффекты

| **Фактор** | **Переменные-эффекты** | | | | |
|---|---|---|---|---|---|
| sp | sp1 $x_1$ | sp2 $x_2$ | sp3 $x_3$ | sp4 $x_4$ | sp5 $x_5$ |
| Крапив | 1 | 0 | 0 | 0 | 0 |
| ЛугКон | 0 | 1 | 0 | 0 | 0 |
| Малин | 0 | 0 | 1 | 0 | 0 |
| БелТряс | 0 | 0 | 0 | 1 | 0 |
| ЛесКон | 0 | 0 | 0 | 0 | 1 |
| ЛесЗав | -1 | -1 | -1 | -1 | -1 |

Переменных-эффектов всегда на одну меньше, чем число уровней фактора.

Переменные закодированы при помощи -1, 0 и 1 (сумма кодов для возможных состояний одной переменной равна нулю).

Для последней группы все переменные-эффекты будут равны $-1$.

# Уравнение модели в параметризации эффектов

| **Фактор** | **Переменные-эффекты** | | | | |
|---|---|---|---|---|---|
| `sp` | `sp1` $x_1$ | `sp2` $x_2$ | `sp3` $x_3$ | `sp4` $x_4$ | `sp5` $x_5$ |
| Крапив | 1 | 0 | 0 | 0 | 0 |
| ЛугКон | 0 | 1 | 0 | 0 | 0 |
| Малин | 0 | 0 | 1 | 0 | 0 |
| БелТряс | 0 | 0 | 0 | 1 | 0 |
| ЛесКон | 0 | 0 | 0 | 0 | 1 |
| ЛесЗав | -1 | -1 | -1 | -1 | -1 |

$$y_i = b_0 + b_1 x_{1i} + ... + b_5 x_{5i} + e_i$$

- $b_0$ — это общее среднее значение отклика.
- $b_1, ..., b_5$ — это отклонения от общего среднего для средних с другими уровнями фактора, кроме последнего.
- для последнего уровня фактора отклонения от общего среднего — это коэффициенты $b_1, ..., b_5$, взятые с противоположным знаком.

# Подбираем коэффициенты модели в параметризации эффектов

```
mod_sum <- lm(len ~ sp, data = cu, contrasts = list(sp = contr.sum))
coef(mod_sum)
```

```
# (Intercept)         sp1         sp2         sp3         sp4         sp5
# 22.50842262 -1.38842262 -0.21508929  0.04782738  0.37824405  0.57157738
```

Коэффициенты моделей будут разными в разных параметризациях, но предсказания будут совершенно одинаковыми.

# Уравнение линейной модели в параметризации эффектов

```
coef(mod_sum)
```

```
# (Intercept)         sp1         sp2         sp3         sp4         sp5
# 22.50842262 -1.38842262 -0.21508929  0.04782738  0.37824405  0.57157738
```

$$\widehat{len}_i = 22.51 - 1.39 sp_{1\ i} - 0.22 sp_{2\ i} + 0.05 sp_{3\ i} + 0.38 sp_{4\ i} + 0.57 sp_{5\ i}$$

Первый коэффициент — средний размер яиц кукушек по всем данным:

- $\overline{len} = 22.51$

Другие коэффициенты — отличие размеров яиц в гнездах хозяев от общего среднего.

Для всех хозяев, кроме последнего, эти отличия будут взяты со знаком "+":

- $\widehat{len}_{\text{Крапив}\ i} = 22.51 - 1.39 sp_{1\ i} = 21.12$
- $\widehat{len}_{\text{ЛугКон}\ i} = 22.51 - 0.22 sp_{2\ i} = 22.29$
- $\widehat{len}_{\text{Малин}\ i} = 22.51 + 0.05 sp_{3\ i} = 22.56$
- $\widehat{len}_{\text{БелТряс}\ i} = 22.51 + 0.38 sp_{4\ i} = 22.89$
- $\widehat{len}_{\text{ЛесКон}\ i} = 22.51 + 0.57 sp_{5\ i} = 23.08$

Для последнего уровня фактора отличия будут взяты со знаком "-", т.к. все переменные-эффекты будут принимать значение -1:

- $\widehat{len}_{\text{ЛесЗав}\ i} = 22.51 - 1.39 sp_{1\ i} - 0.22 sp_{2\ i} + 0.05 sp_{3\ i} + 0.38 sp_{4\ i} + 0.57 sp_{5\ i} = 23.12$

# t-тесты значимости коэффициентов

# t-тесты значимости коэффициентов

- Для модели в параметризации индикаторов t-тесты угловых коэффициентов показывают значимость отличий средних значений в группах от среднего на базовом уровне.
- По значениям коэффициентов нельзя сказать влияет ли дискретный фактор целиком (исключение — фактор с двумя градациями).

```
coef(summary(mod_treatment))
```

```
#              Estimate Std. Error   t value      Pr(>|t|)
# (Intercept) 21.120000  0.2337213 90.364038 6.199539e-108
# spЛугКон     1.173333  0.2698781  4.347642  3.006702e-05
# spМалин      1.436250  0.3253263  4.414799  2.309832e-05
# spБелТряс    1.766667  0.3305318  5.344922  4.699402e-07
# spЛесКон     1.960000  0.3305318  5.929837  3.309942e-08
# spЛесЗав     1.994286  0.3363824  5.928627  3.328637e-08
```

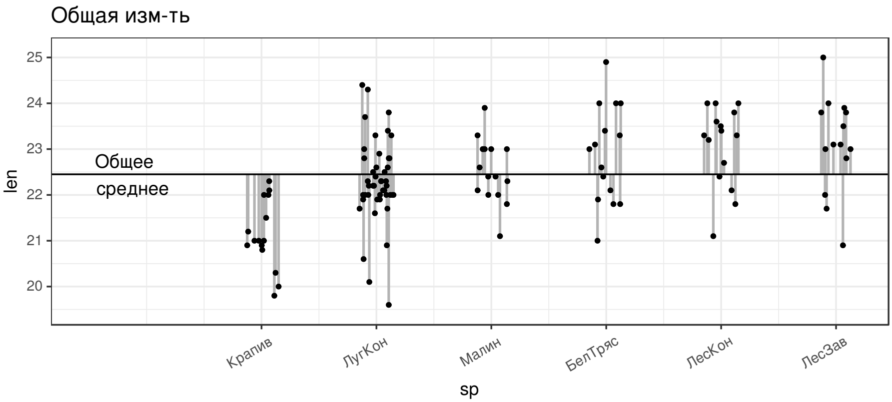
Для модели в параметризации эффектов t-тесты угловых коэффициентов показывают значимость отличий средних в группах от общего среднего — такое сравнение редко имеет смысл.

```
coef(summary(mod_sum))
```

```
#                Estimate Std. Error     t value      Pr(>|t|)
# (Intercept) 22.50842262 0.09003464 249.9973693 5.090356e-158
# sp1         -1.38842262 0.21100553  -6.5800297  1.492281e-09
# sp2         -0.21508929 0.14228587  -1.5116701  1.333850e-01
# sp3          0.04782738 0.20554139   0.2326898  8.164196e-01
# sp4          0.37824405 0.21100553   1.7925789  7.569241e-02
# sp5          0.57157738 0.21100553   2.7088266  7.793598e-03
```

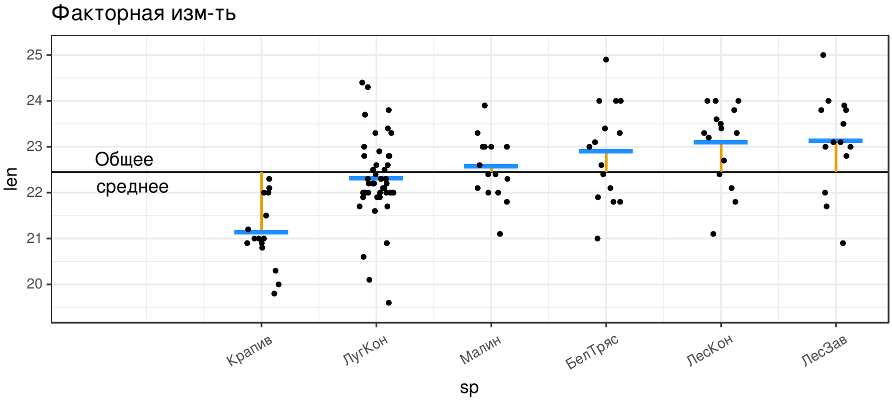
# Дисперсионный анализ

# Общая изменчивость




Общая изменчивость $\mathrm{SS_t}$ — это сумма квадратов отклонений наблюдаемых значений $y_i$ от общего среднего $\bar{y}$

# Факторная (межгрупповая) изменчивость

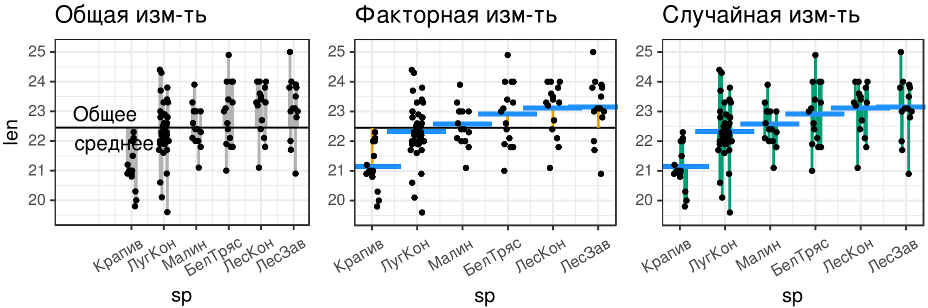



Отклонения внутригрупповых средних от общего среднего в генеральной совокупности — это эффект фактора $\alpha_j = \mu_j - \mu$, где $j = 1, 2, ..., p$ — это одна из $p$ групп.

Мы оцениваем эффект фактора по реальным данным $\bar{y}_j - \bar{y}$

# Структура общей изменчивости

$$SS_t = SS_x + SS_e$$



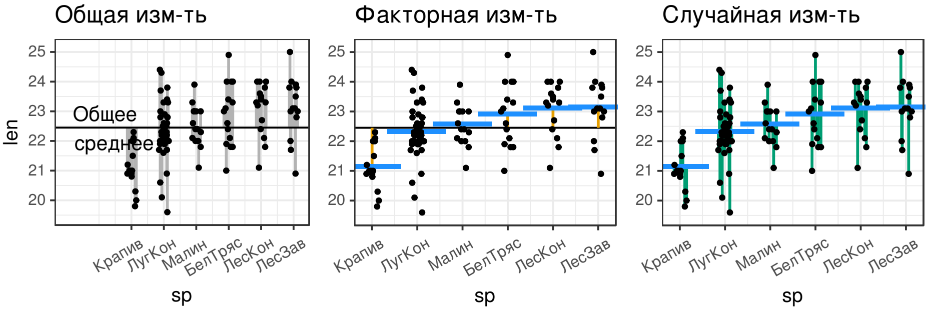

$$SS_t = \sum\sum(\bar{y} - y_{ij})^2$$
$$df_t = n - 1$$

$$SS_x = \sum n_j(\bar{y}_j - \bar{y})^2$$
$$df_x = p - 1$$

$$SS_e = \sum\sum(\bar{y}_j - y_{ij})^2$$
$$df_e = n - p$$

# От изменчивостей к дисперсиям

$$SS_t = SS_x + SS_e \qquad MS_t \neq MS_x + MS_e$$




$$SS_t = \sum \sum (\bar{y} - y_{ij})^2$$
$$df_t = n - 1$$
$$MS_t = \frac{SS_t}{df_t}$$

$$SS_x = \sum n_j (\bar{y}_j - \bar{y})^2$$
$$df_x = p - 1$$
$$MS_x = \frac{SS_x}{df_x}$$

$$SS_e = \sum \sum (\bar{y}_j - y_{ij})^2$$
$$df_e = n - p$$
$$MS_e = \frac{SS_e}{df_e}$$

# $MS_x$ и $MS_e$ помогают тестировать значимость фактора

Если дисперсии остатков в группах равны и фактор имеет фиксированное число градаций:

$$E(MS_x) = \sigma^2 + \sum n_i \frac{(\mu_i - \mu)^2}{p-1} = \sigma^2 + \sigma_x^2$$

$$E(MS_e) = \sigma^2$$

# $MS_x$ и $MS_e$ помогают тестировать значимость фактора

Если дисперсии остатков в группах равны и фактор имеет фиксированное число градаций:

$$E(MS_x) = \sigma^2 + \sum n_i \frac{(\mu_i - \mu)^2}{p-1} = \sigma^2 + \sigma_x^2$$

$$E(MS_e) = \sigma^2$$

Если зависимости нет, то $\mu_1 = ... = \mu_p$ — средние равны во всех $p$ группах, и тогда $MS_x \sim MS_e$.

# $MS_x$ и $MS_e$ помогают тестировать значимость фактора

Если дисперсии остатков в группах равны и фактор имеет фиксированное число градаций:

$E(MS_x) = \sigma^2 + \sum n_i \frac{(\mu_i - \mu)^2}{p-1} = \sigma^2 + \sigma_x^2$

$E(MS_e) = \sigma^2$

Если зависимости нет, то $\mu_1 = ... = \mu_p$ — средние равны во всех $p$ группах, и тогда $MS_x \sim MS_e$.

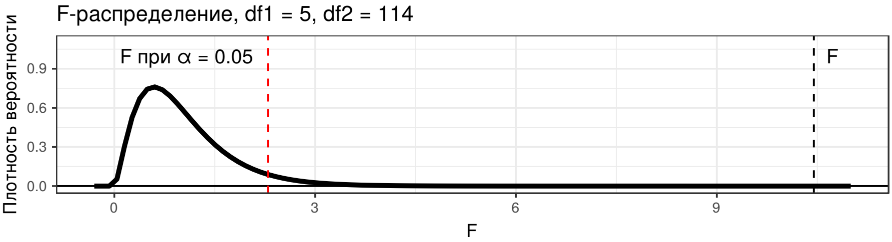
- $H_0 : \mu_1 = ... = \mu_p$ — средние во всех $p$ группах равны.
- $H_A : \exists\ i, j : \mu_i \neq \mu_j$ — **хотя бы одно** среднее отличается от общего среднего.

$$F_{df_x, df_e} = \frac{MS_x}{MS_e}$$

# Тестирование значимости фактора при помощи F-критерия

$$F_{df_x, df_e} = \frac{MS_x}{MS_e}$$

В однофакторном дисперсионном анализе $df_x = p - 1$ и $df_e = n - p$.



# Результаты дисперсионного анализа часто представляют в виде таблицы

| **Источник изменчивости** | **SS** | **df** | **MS** | **F** |
|---|---|---|---|---|
| Название фактора | $SS_x = \sum n_j(\bar{y}_j - \bar{y})^2$ | $df_x = p - 1$ | $MS_x = \frac{SS_x}{df_x}$ | $F_{df_x df_e} = \frac{MS_x}{MS_e}$ |
| Случайная | $SS_e = \sum\sum(\bar{y}_j - y_{ij})^2$ | $df_e = n - p$ | $MS_e = \frac{SS_e}{df_e}$ | |
| Общая | $SS_t = \sum\sum(\bar{y} - y_{ij})^2$ | $df_t = n - 1$ | | |

Минимальное описание результатов в тексте должно содержать $F_{df_x, df_e}$ и $p$.

# Делаем дисперсионный анализ в R

В R есть много функций для дисперсионного анализа. Мы рекомендуем `Anova()` (**с большой буквы**) из пакета `car`. Зачем? Эта функция умеет тестировать влияние факторов в определенном порядке. Когда факторов будет больше одного, это станет важно для результатов.

```
library(car)
cu_anova <- Anova(mod_treatment)
cu_anova
```

```
# Anova Table (Type II tests)
#
# Response: len
#           Sum Sq  Df F value        Pr(>F)
# sp         42.81   5  10.449 0.00000002852 ***
# Residuals  93.41 114
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Результаты дисперсионного анализа

Результаты дисперсионного анализа можно описать в тексте:

- Длина яиц кукушек в гнездах разных птиц-хозяев значимо различается ($F_{5,114} = 10.45$, $p < 0.01$).

# Результаты дисперсионного анализа

Результаты дисперсионного анализа можно представить в виде таблицы

- Длина яиц кукушек значимо различалась в гнездах разных птиц-хозяев (Табл. 1).

Table 1: Результаты дисперсионного анализа длины яиц кукушек в гнездах разных птиц-хозяев. SS — суммы квадратов отклонений, df — число степеней свободы, F — значение F-критерия, P — доверительная вероятность.

| | SS | df | F | P |
|---|---|---|---|---|
| Хозяин | 42.8 | 5 | 10.4 | $< 0.01$ |
| Остаточная | 93.4 | 114 | | |

# Условия примененимости дисперсионного анализа

# Результатам тестов можно верить, если выполняются условия применимости

Условия применимости дисперсионного анализа:

- Случайность и независимость наблюдений внутри групп
- Нормальное распределение остатков
- Гомогенность дисперсий остатков
- Отсутствие коллинеарности факторов (независимость групп)

## Другие ограничения:

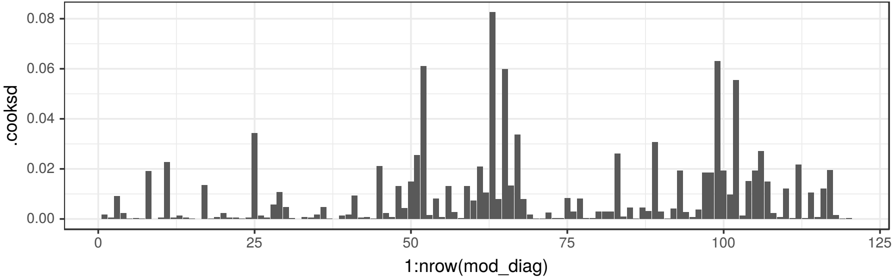
- Лучше работает, если размеры групп примерно одинаковы (т.наз. сбалансированный дисперсионный комплекс)
- Устойчив к отклонениям от нормального распределения (при равных объемах групп или при больших выборках)

# Проверяем выполнение условий применимости

```
# Данные для графиков остатков
mod_diag <- fortify(mod_treatment)
```

## 1) График расстояния Кука

```
ggplot(mod_diag, aes(x = 1:nrow(mod_diag), y = .cooksd)) +
  geom_bar(stat = "identity")
```




- Выбросов нет

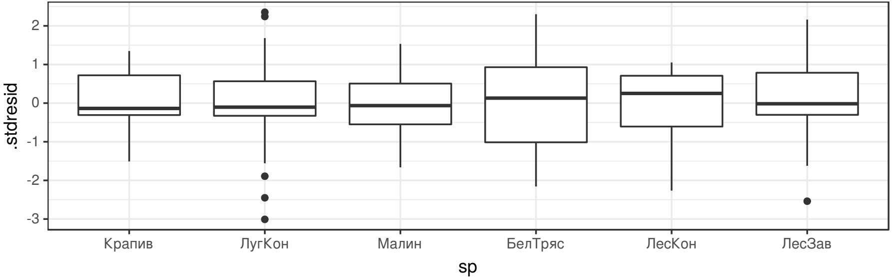
## 2) График остатков от предсказанных значений

```
ggplot(mod_diag, aes(x = .fitted, y = .stdresid)) + geom_jitter()
```

У нас один единственный дискретный предиктор, поэтому удобнее сразу боксплот

## 3) Графики остатков от предикторов в модели и не в модели

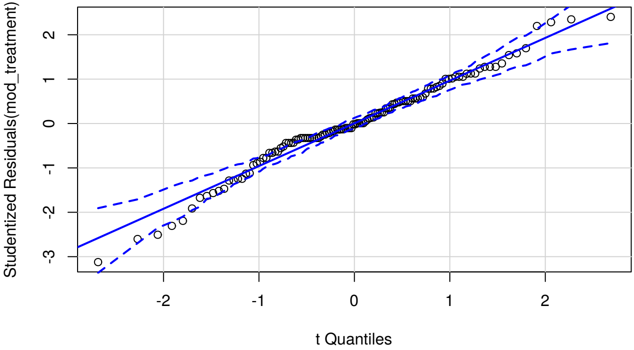
```
ggplot(mod_diag, aes(x = sp, y = .stdresid)) + geom_boxplot()
```




- Дисперсии почти одинаковые. Может быть, в одной из групп чуть больше

# 4) Квантильный график остатков

```
library(car)
qqPlot(mod_treatment, id = FALSE)
```




- Распределение остатков немного отличается от нормального

# Пост хок тесты

# Как понять, какие именно группы различаются

Дисперсионный анализ говорит нам только, есть ли влияние фактора, но не говорит, какие именно группы различаются.

Коэффициенты линейной модели в `summary(mod_treatment)` содержат лишь часть ответа — сравнение средних значених всех групп со средним на базовом уровне.

Если нас интересуют другие возможные попарные сравнения, нужно сделать пост хок тест.

# Post hoc тесты

Пост хок тесты — попарные сравнения средних **после того, как дисперсионный анализ показал, что влияние фактора достоверно**

## Свойства post hoc тестов:

- **Применяются, только если влияние фактора значимо**
- Делают поправку для снижения вероятности ошибки I рода $\alpha$, (но не слишком большую, чтобы не снизилась мощность, и чтобы не возросла вероятность ошибки II рода $\beta$)
    - Учитывают величину различий между средними
    - Учитывают количество сравниваемых пар
- Различаются по степени консервативности (тест Тьюки — разумный компромисс)
- Работают лучше при равных объемах групп, при гомогенности дисперсий

# Пост хок тест Тьюки в R

- `glht()` — “general linear hypotheses testing”
- `linfct` — аргумент, задающий гипотезу для тестирования
- `mcp()` — функция, чтобы задавать множественные сравнения (обычные пост хоки)
- `sp = “Tukey”` — тест Тьюки по фактору `sp`

```
library(multcomp)
cu_ph <- glht(mod_treatment, linfct = mcp(sp = "Tukey"))
```

# Результаты попарных сравнений (тест Тьюки)

Таблица результатов пост хок теста практически нечитабельна.

```
summary(cu_ph)
```

```
#
#    Simultaneous Tests for General Linear Hypotheses
#
# Multiple Comparisons of Means: Tukey Contrasts
#
#
# Fit: lm(formula = len ~ sp, data = cu)
#
# Linear Hypotheses:
#                          Estimate Std. Error t value Pr(>|t|)
# ЛугКон - Крапив == 0      1.17333    0.26988   4.348   <0.001 ***
# Малин - Крапив == 0       1.43625    0.32533   4.415   <0.001 ***
# БелТряс - Крапив == 0     1.76667    0.33053   5.345   <0.001 ***
# ЛесКон - Крапив == 0      1.96000    0.33053   5.930   <0.001 ***
# ЛесЗав - Крапив == 0      1.99429    0.33638   5.929   <0.001 ***
# Малин - ЛугКон == 0       0.26292    0.26348   0.998   0.9153
# БелТряс - ЛугКон == 0     0.59333    0.26988   2.199   0.2415
# ЛесКон - ЛугКон == 0      0.78667    0.26988   2.915   0.0466 *
# ЛесЗав - ЛугКон == 0      0.82095    0.27701   2.964   0.0409 *
# БелТряс - Малин == 0      0.33042    0.32533   1.016   0.9093
# ЛесКон - Малин == 0       0.52375    0.32533   1.610   0.5870
# ЛесЗав - Малин == 0       0.55804    0.33127   1.685   0.5378
# ЛесКон - БелТряс == 0     0.19333    0.33053   0.585   0.9916
# ЛесЗав - БелТряс == 0     0.22762    0.33638   0.677   0.9836
# ЛесЗав - ЛесКон == 0      0.03429    0.33638   0.102   1.0000
# ---
```

# Результаты пост хок теста

Результаты пост хок теста можно привести в виде текста…

- Размер яиц кукушек в гнездах крапивника значимо меньше, чем в гнездах лугового конька (тест Тьюки, $p < 0.01$). Размер яиц кукушек в гнездах лесной завирушки, белой трясогузки, малиновки и лесного конька не различается, но яйца кукушек в гнездах этих хозяев крупнее, чем в гнездах у лугового конька или крапивника (тест Тьюки, от $p < 0.01$ до $0.05$).

…или построить график

# Данные для графика при помощи predict()

```
MyData <- data.frame(sp = factor(levels(cu$sp), levels = levels(cu$sp)))

MyData <- data.frame(
  MyData,
  predict(mod_treatment, newdata = MyData, interval = "confidence"))

MyData
```

```
#        sp      fit      lwr      upr
# 1  Крапив 21.12000 20.65700 21.58300
# 2  ЛугКон 22.29333 22.02602 22.56065
# 3   Малин 22.55625 22.10795 23.00455
# 4 БелТряс 22.88667 22.42367 23.34967
# 5  ЛесКон 23.08000 22.61700 23.54300
# 6  ЛесЗав 23.11429 22.63504 23.59354
```

# Задание

Создайте MyData вручную:

- предсказанные значения
- стандартные ошибки
- верхнюю и нижнюю границы доверительных интервалов

```
MyData <- data.frame(sp = factor(levels(cu$sp), levels = levels(cu$sp)))

X <- model.matrix()
betas <-
MyData$fit <-  %*%
MyData$se <- sqrt(diag(X %*% vcov(mod_treatment) %*% t(X)))
t_crit <- qt(p = , df = nrow() - length(coef()))
MyData$lwr <- MyData$ -  * MyData$
MyData$upr <- MyData$ +  * MyData$
```

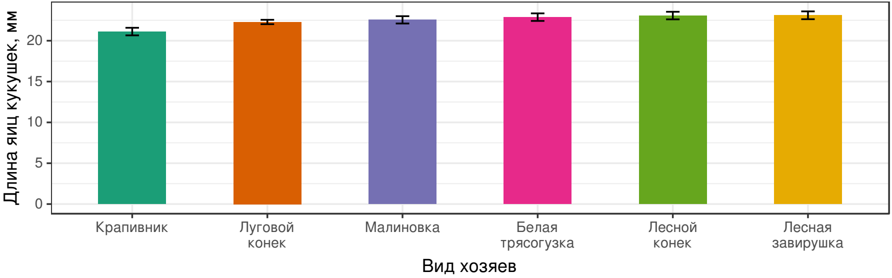
# Решение:

```
MyData <- data.frame(sp = factor(levels(cu$sp), levels = levels(cu$sp)))
X <- model.matrix(~sp, data = MyData)
betas <- coef(mod_treatment)
MyData$fit <- X %*% betas
MyData$se <- sqrt(diag(X %*% vcov(mod_treatment) %*% t(X)))
t_crit <- qt(p = 0.975, df = nrow(cu) - length(coef(mod_treatment)))
MyData$lwr <- MyData$fit - t_crit * MyData$se
MyData$upr <- MyData$fit + t_crit * MyData$se
MyData
```

```
#        sp      fit        se      lwr      upr
# 1  Крапив 21.12000 0.2337213 20.65700 21.58300
# 2  ЛугКон 22.29333 0.1349391 22.02602 22.56065
# 3   Малин 22.55625 0.2262997 22.10795 23.00455
# 4 БелТряс 22.88667 0.2337213 22.42367 23.34967
# 5  ЛесКон 23.08000 0.2337213 22.61700 23.54300
# 6  ЛесЗав 23.11429 0.2419245 22.63504 23.59354
```

# Столбчатый график

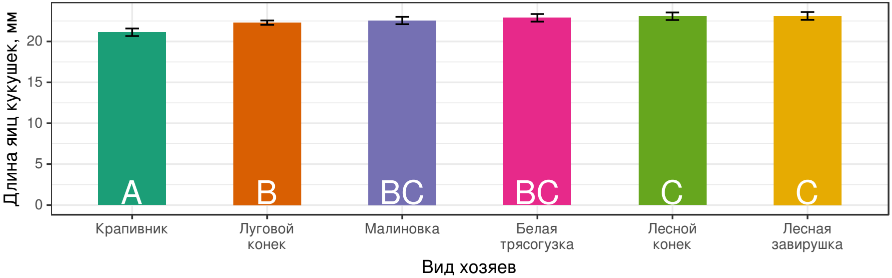
```
gg_bars <- ggplot(data = MyData, aes(x = sp, y = fit)) +
  geom_bar(stat = "identity", aes(fill = sp), width = 0.5) +
  geom_errorbar(aes(ymin = lwr, ymax = upr), width = 0.1) +
  labs(x = "Вид хозяев", y = "Длина яиц кукушек, мм") +
  scale_fill_brewer(name = "Вид \nхозяев", palette = "Dark2") +
  scale_x_discrete(labels = c("Крапивник", "Луговой\nконек", "Малиновка",
                              "Белая\nтрясогузка", "Лесной\nконек", "Лесная\nзавирушка"
  theme(legend.position = "none")
gg_bars
```



# Можно привести результаты пост хок теста на столбчатом графике

Значимо различающиеся группы обозначим разными буквами

```
gg_bars_coded <- gg_bars +
  geom_text(aes(y = 1.6,  label = c("A", "B", "BC", "BC", "C", "C")),
            colour = "white", size = 7)
gg_bars_coded
```



# Take home messages

- Дисперсионный анализ — линейная модель с дискретными предикторами, существует в нескольких параметризациях, которые отличаются трактовками коэффициентов
- При помощи дисперсионного анализа можно проверить гипотезу о равенстве средних значений в группах
- Условия применимости дисперсионного анализа
  - Случайность и независимость групп и наблюдений внутри групп
  - Нормальное распределение в группах
  - Гомогенность дисперсий в группах
- При множественных попарных сравнениях увеличивается вероятность ошибки первого рода, поэтому нужно вносить поправку для уровня значимости
- Post hoc тесты — это попарные сравнения после дисперсионного анализа, которые позволяют сказать, какие именно средние различаются

# Дополнительные ресурсы

- Quinn, Keough, 2002, pp. 173–207
- Logan, 2010, pp. 254–282
- Open Intro to Statistics, pp.236–246
- Sokal, Rohlf, 1995, pp. 179–260
- Zar, 2010, pp. 189-207