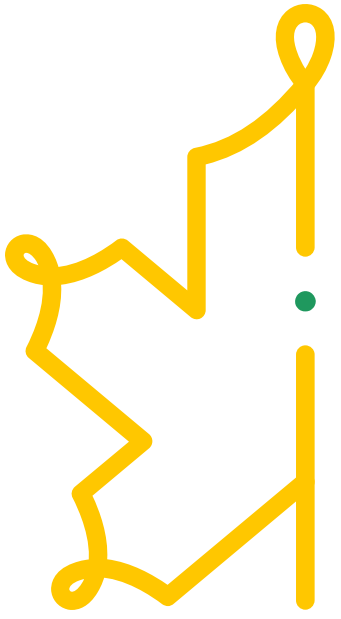
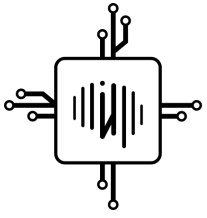


Razвед
анализ
данных

Работа с табличными данными

Продвинутый dplyr



Фонд
интеллект

Анна Валяева

Лекция 10 - 2022

Работа с табличными данными

Работа с табличными данными

Очень широкий датафрейм, который мы создали на прошлом семинаре.

Как привести его к форме `name-weight_1-weight_2-weight_3`?

```
lemurs_weights_wide
```

```
# A tibble: 3 x 52
  weight_date Agatha Angelique `Annabel Lee` `Ardrey-A` Ardrey `Bellatrix-A`
  <chr>        <dbl>     <dbl>         <dbl>      <dbl>  <dbl> <lgl>
1 weight_1      1060      2920           944         98   3000 NA
2 weight_2      1860      2940          1180         98   2780 NA
3 weight_3      2000       209          1689         95    666 NA
# ... with 45 more variables: Bellatrix-B <lgl>, Bellatrix-C <lgl>,
#   Bellatrix <dbl>, Blue Devil <dbl>, Caliban <dbl>, Claudia <dbl>,
#   Cruella <dbl>, Damien <lgl>, Elphaba <dbl>, Endora <dbl>, Goblin <dbl>,
#   Grendel <dbl>, Hitchcock <dbl>, Ichabod <dbl>, Imp <lgl>, Kali <dbl>,
#   Kambana <lgl>, Loki <lgl>, Lucrezia <dbl>, Medea <dbl>, Medusa <dbl>,
#   Mephistopheles <dbl>, Merlin <dbl>, Morticia <dbl>, Niffy <lgl>,
#   Norman Bates <dbl>, Nosferatu <dbl>, Ozma-A <dbl>, Ozma <dbl>, ...
```

Работа с табличными данными

Как привести его к форме `name-weight_1-weight_2-weight_3`?

```
# A tibble: 51 x 4
   name        weight_1 weight_2 weight_3
   <chr>          <dbl>    <dbl>    <dbl>
 1 Agatha          1060     1860     2000
 2 Angelique       2920     2940      209
 3 Annabel Lee      944     1180     1689
 4 Ardrey-A          98       98       95
 5 Ardrey          3000     2780      666
 6 Bellatrix-A       NA       NA       NA
 7 Bellatrix-B       NA       NA       NA
 8 Bellatrix-C       NA       NA       NA
 9 Bellatrix        585     2760     2460
10 Blue Devil      1330     1820     2460
# ... with 41 more rows
```

Работа с табличными данными

Как привести его к форме `name-weight_1-weight_2-weight_3`?

```
lemurs_weights <- lemurs_weights_wide %>%
  pivot_longer(-weight_date) %>%
  pivot_wider(names_from = weight_date, values_from = value)

lemurs_weights
```

```
# A tibble: 51 x 4
   name        weight_1 weight_2 weight_3
   <chr>          <dbl>    <dbl>    <dbl>
 1 Agatha          1060     1860     2000
 2 Angelique       2920     2940      209
 3 Annabel Lee      944     1180     1689
 4 Ardrey-A          98       98       95
 5 Ardrey          3000     2780      666
 6 Bellatrix-A       NA       NA       NA
 7 Bellatrix-B       NA       NA       NA
 8 Bellatrix-C       NA       NA       NA
 9 Bellatrix        585     2760     2460
10 Blue Devil      1330     1820     2460
# ... with 41 more rows
```

Как еще можно указать множество столбцов?

- Использовать информацию о типе данных: `where(is.character)`, ...

```
lemurs_weights_wide %>%
  pivot_longer(!where(is.character)) %>%
  pivot_wider(names_from = weight_date, values_from = value)

lemurs_weights_wide %>%
  pivot_longer(where(is.logical) | where(is.numeric)) %>%
  pivot_wider(names_from = weight_date, values_from = value)
```

Как еще можно указать множество столбцов?

- `starts with("pattern")` - начинается с "pattern"
- `ends with("pattern")` - заканчивается на "pattern"
- `contains("pattern")` - содержит подслово "pattern"
- `matches("pattern")` - находится по регулярному выражению "pattern"

```
lemurs_weights %>% select(starts_with("weight")) %>% head(1)
```

```
# A tibble: 1 x 3
  weight_1 weight_2 weight_3
     <dbl>    <dbl>    <dbl>
1     1060     1860     2000
```

```
lemurs_weights %>% select(matches("*_[12]")) %>% head(1)
```

```
# A tibble: 1 x 2
  weight_1 weight_2
     <dbl>    <dbl>
1     1060     1860
```

Как еще можно указать множество столбцов?

- `num_range()` - поиск по общему префиксу среди столбцов с некой нумерацией

```
lemurs_weights %>%
  select(num_range("weight_", c(1,3))) %>% # prefix, numeric range
  head(1)
```

```
# A tibble: 1 x 2
  weight_1 weight_3
     <dbl>    <dbl>
1     1060     2000
```

Как еще можно указать множество столбцов?

- Использовать информацию о позиции столбца

```
lemurs_weights %>% select(1, num_range("weight_", c(1,3))) %>% head(1)
```

```
# A tibble: 1 x 3
  name   weight_1 weight_3
  <chr>     <dbl>    <dbl>
1 Agatha     1060     2000
```

Как еще можно указать множество столбцов?

- Добавить условие по значениям в столбцах

```
lemurs_weights %>%
  select(where(~ is.numeric(.) && max(., na.rm=TRUE) > 3000)) %>%
  head(1)
```

```
# A tibble: 1 x 1
  weight_3
     <dbl>
1     2000
```

Устаревшее:

```
lemurs_weights %>%
  select_if(~ is.numeric(.) && max(., na.rm=TRUE) > 3000)
```

Как еще можно указать множество столбцов?

- Использовать вектор с названиями нужных столбцов и `all_of()` или `any_of()`.

```
weight_cols <- paste("weight", 1:4, sep = "_")
```

```
lemurs_weights %>%
  select(all_of(weight_cols))
```

```
Error: Can't subset columns that don't
   exist.
x Column `weight_4` doesn't exist.
```

```
lemurs_weights %>%
  select(any_of(weight_cols)) %>%
  head(1)
```

```
# A tibble: 1 x 3
  weight_1 weight_2 weight_3
     <dbl>    <dbl>    <dbl>
1     1060     1860     2000
```

Трансформация таблиц

Задача: по 3 взвешиваниям посчитать средний вес каждого лемура.

```
lemurs_weights
```

```
# A tibble: 51 x 4
   name        weight_1 weight_2 weight_3
   <chr>          <dbl>    <dbl>    <dbl>
 1 Agatha          1060     1860     2000
 2 Angelique       2920     2940      209
 3 Annabel Lee      944     1180     1689
 4 Ardrey-A          98       98       95
 5 Ardrey          3000     2780      666
 6 Bellatrix-A       NA       NA       NA
 7 Bellatrix-B       NA       NA       NA
 8 Bellatrix-C       NA       NA       NA
 9 Bellatrix        585     2760     2460
10 Blue Devil      1330     1820     2460
# ... with 41 more rows
```

Подсчет по нескольким столбцам

Задача: по 3 взвешиваниям посчитать средний вес каждого лемура.

Получилось что-то странное...

```
lemurs_weights %>%
  mutate(
    avg_weight = mean(weight_1:weight_3))
```

```
# A tibble: 51 x 5
   name        weight_1 weight_2 weight_3 avg_weight
   <chr>          <dbl>    <dbl>    <dbl>      <dbl>
 1 Agatha          1060     1860     2000       1530
 2 Angelique       2920     2940      209       1530
 3 Annabel Lee      944     1180     1689       1530
 4 Ardrey-A          98       98       95       1530
 5 Ardrey          3000     2780      666       1530
 6 Bellatrix-A       NA       NA       NA       1530
 7 Bellatrix-B       NA       NA       NA       1530
 8 Bellatrix-C       NA       NA       NA       1530
 9 Bellatrix        585     2760     2460       1530
10 Blue Devil      1330     1820     2460       1530
# ... with 41 more rows
```

Подсчет по нескольким столбцам

Задача: по 3 взвешиваниям посчитать средний вес каждого лемура.

Получилось верно, но как-то глупо считать среднее "вручную"...

```
lemurs_weights %>%
  mutate(
    avg_weight = (weight_1 + weight_2 + weight_3) / 3)
```

```
# A tibble: 51 x 5
   name        weight_1 weight_2 weight_3 avg_weight
   <chr>          <dbl>    <dbl>    <dbl>      <dbl>
 1 Agatha          1060     1860     2000      1640
 2 Angelique       2920     2940      209      2023
 3 Annabel Lee      944     1180     1689      1271
 4 Ardrey-A          98       98       95        97
 5 Ardrey          3000     2780      666      2149.
 6 Bellatrix-A       NA       NA       NA        NA
 7 Bellatrix-B       NA       NA       NA        NA
 8 Bellatrix-C       NA       NA       NA        NA
 9 Bellatrix        585     2760     2460      1935
10 Blue Devil      1330     1820     2460      1870
# ... with 41 more rows
```

Подсчет по нескольким столбцам

Задача: по 3 взвешиваниям посчитать средний вес каждого лемура.

Группируем **построчно** и для каждой строки считаем среднее. Каждый лемур сам себе группа.

```
lemurs_weights %>%
  rowwise() %>%
  mutate(
    avg_weight = mean(c(weight_1, weight_2, weight_3), na.rm = TRUE))
```

```
# A tibble: 51 x 5
# Rowwise:
   name        weight_1 weight_2 weight_3 avg_weight
   <chr>          <dbl>    <dbl>    <dbl>      <dbl>
 1 Agatha          1060     1860     2000      1640
 2 Angelique       2920     2940      209      2023
 3 Annabel Lee      944     1180     1689      1271
 4 Ardrey-A          98       98       95        97
 5 Ardrey          3000     2780      666      2149.
 6 Bellatrix-A       NA       NA       NA       NaN
 7 Bellatrix-B       NA       NA       NA       NaN
 8 Bellatrix-C       NA       NA       NA       NaN
 9 Bellatrix        585     2760     2460      1935
10 Blue Devil      1330     1820     2460      1870
# ... with 41 more rows
```

Подсчет по нескольким столбцам

Задача: по 3 взвешиваниям посчитать средний вес каждого лемура.

`c_across()` позволяет отбирать столбцы по-умному (как срез, по типу данных, ...).

```
lemurs_weights %>%
  rowwise() %>%
  mutate(
    avg_weight = mean(c_across(weight_1:weight_3), na.rm = TRUE))
```

```
# A tibble: 51 x 5
# Rowwise:
   name        weight_1 weight_2 weight_3 avg_weight
   <chr>          <dbl>    <dbl>    <dbl>      <dbl>
 1 Agatha          1060     1860     2000      1640
 2 Angelique       2920     2940      209      2023
 3 Annabel Lee      944     1180     1689      1271
 4 Ardrey-A          98       98       95        97
 5 Ardrey          3000     2780      666      2149.
 6 Bellatrix-A       NA       NA       NA       NaN
 7 Bellatrix-B       NA       NA       NA       NaN
 8 Bellatrix-C       NA       NA       NA       NaN
 9 Bellatrix        585     2760     2460      1935
10 Blue Devil      1330     1820     2460      1870
# ... with 41 more rows
```

Подсчет по нескольким столбцам

Задача: по 3 взвешиваниям посчитать средний вес каждого лемура.

`c_across()` позволяет отбирать столбцы по-умному (как срез, по типу данных, ...).

```
lemurs_weights %>%
  rowwise() %>%
  mutate(
    avg_weight = mean(c_across(where(is.numeric)), na.rm = TRUE))
```

```
# A tibble: 51 x 5
# Rowwise:
   name        weight_1 weight_2 weight_3 avg_weight
   <chr>          <dbl>    <dbl>    <dbl>      <dbl>
 1 Agatha          1060     1860     2000      1640
 2 Angelique       2920     2940      209      2023
 3 Annabel Lee      944     1180     1689      1271
 4 Ardrey-A          98       98       95        97
 5 Ardrey          3000     2780      666      2149.
 6 Bellatrix-A       NA       NA       NA       NaN
 7 Bellatrix-B       NA       NA       NA       NaN
 8 Bellatrix-C       NA       NA       NA       NaN
 9 Bellatrix        585     2760     2460      1935
10 Blue Devil      1330     1820     2460      1870
# ... with 41 more rows
```

Подсчет по нескольким столбцам

`rowwise()` создает группировку, которую умеет снимать `summarise()` или `ungroup()`.

```
lemurs_weights %>%
  rowwise() %>%
  mutate(
    avg_weight = mean(c_across(where(is.numeric)), na.rm = TRUE)) %>%
  ungroup()
```

```
# A tibble: 51 x 5
   name        weight_1 weight_2 weight_3 avg_weight
   <chr>          <dbl>    <dbl>    <dbl>      <dbl>
 1 Agatha          1060     1860     2000      1640
 2 Angelique       2920     2940      209      2023
 3 Annabel Lee      944     1180     1689      1271
 4 Ardrey-A          98       98       95        97
 5 Ardrey          3000     2780      666      2149.
 6 Bellatrix-A       NA       NA       NA       NaN
 7 Bellatrix-B       NA       NA       NA       NaN
 8 Bellatrix-C       NA       NA       NA       NaN
 9 Bellatrix        585     2760     2460      1935
10 Blue Devil      1330     1820     2460      1870
# ... with 41 more rows
```

Трансформировать сразу все столбцы

Столбцы перезаписываются.

```
lemurs_weights %>%
  mutate(across(everything(), toupper))
```

```
# A tibble: 51 x 4
   name        weight_1 weight_2 weight_3
   <chr>       <chr>    <chr>    <chr>
 1 AGATHA      1060     1860     2000
 2 ANGELIQUE   2920     2940     209
 3 ANNABEL LEE 944      1180     1689
 4 ARDREY-A    98       98       95
 5 ARDREY      3000     2780     666
 6 BELLATRIX-A <NA>     <NA>     <NA>
 7 BELLATRIX-B <NA>     <NA>     <NA>
 8 BELLATRIX-C <NA>     <NA>     <NA>
 9 BELLATRIX   585      2760     2460
10 BLUE DEVIL  1330     1820     2460
# ... with 41 more rows
```

Трансформировать несколько столбцов

```
lemurs_weights %>%
  mutate(across(c("name"), toupper))
```

```
# A tibble: 51 x 4
   name        weight_1 weight_2 weight_3
   <chr>          <dbl>    <dbl>    <dbl>
 1 AGATHA          1060     1860     2000
 2 ANGELIQUE       2920     2940      209
 3 ANNABEL LEE      944     1180     1689
 4 ARDREY-A          98       98       95
 5 ARDREY          3000     2780      666
 6 BELLATRIX-A       NA       NA       NA
 7 BELLATRIX-B       NA       NA       NA
 8 BELLATRIX-C       NA       NA       NA
 9 BELLATRIX        585     2760     2460
10 BLUE DEVIL      1330     1820     2460
# ... with 41 more rows
```

Трансформировать несколько столбцов

```
lemurs_weights %>%
  mutate(across(where(is.numeric), round))
```

```
# A tibble: 51 x 4
   name        weight_1 weight_2 weight_3
   <chr>          <dbl>    <dbl>    <dbl>
 1 Agatha          1060     1860     2000
 2 Angelique       2920     2940      209
 3 Annabel Lee      944     1180     1689
 4 Ardrey-A          98       98       95
 5 Ardrey          3000     2780      666
 6 Bellatrix-A       NA       NA       NA
 7 Bellatrix-B       NA       NA       NA
 8 Bellatrix-C       NA       NA       NA
 9 Bellatrix        585     2760     2460
10 Blue Devil      1330     1820     2460
# ... with 41 more rows
```

Трансформировать несколько столбцов

```
lemurs_weights %>%
  mutate(across(starts_with("weight"), ~ .x/1000))
```

```
# A tibble: 51 x 4
   name        weight_1 weight_2 weight_3
   <chr>          <dbl>    <dbl>    <dbl>
 1 Agatha         1.06     1.86     2
 2 Angelique      2.92     2.94     0.209
 3 Annabel Lee    0.944    1.18     1.69
 4 Ardrey-A       0.098    0.098    0.095
 5 Ardrey         3        2.78     0.666
 6 Bellatrix-A   NA       NA       NA
 7 Bellatrix-B   NA       NA       NA
 8 Bellatrix-C   NA       NA       NA
 9 Bellatrix      0.585    2.76     2.46
10 Blue Devil     1.33     1.82     2.46
# ... with 41 more rows
```

Трансформировать несколько столбцов

При использовании `list(...)` или при указании `.names = ...` создаются новые столбцы.

```
lemurs_weights %>%
  mutate(across(starts_with("weight"), list(kg = ~ .x/1000)))
```

```
# A tibble: 51 x 7
   name        weight_1 weight_2 weight_3 weight_1_kg weight_2_kg weight_3_kg
   <chr>          <dbl>    <dbl>    <dbl>       <dbl>       <dbl>       <dbl>
 1 Agatha          1060     1860     2000       1.06        1.86        2
 2 Angelique       2920     2940      209       2.92        2.94        0.209
 3 Annabel Lee      944     1180     1689       0.944       1.18        1.69
 4 Ardrey-A          98       98       95       0.098       0.098       0.095
 5 Ardrey          3000     2780      666       3           2.78        0.666
 6 Bellatrix-A       NA       NA       NA      NA          NA          NA
 7 Bellatrix-B       NA       NA       NA      NA          NA          NA
 8 Bellatrix-C       NA       NA       NA      NA          NA          NA
 9 Bellatrix        585     2760     2460       0.585       2.76        2.46
10 Blue Devil      1330     1820     2460       1.33        1.82        2.46
# ... with 41 more rows
```

Переименовать несколько столбцов

```
lemurs_weights %>%
  mutate(across(starts_with("weight"), list(kg = ~ .x/1000))) %>%
  # как модифицировать названия столбцов, какие столбцы
  rename_with(~ str_c("KG_", str_remove(., "_kg")), ends_with("kg"))
```

```
# A tibble: 51 x 7
   name        weight_1 weight_2 weight_3 KG_weight_1 KG_weight_2 KG_weight_3
   <chr>          <dbl>    <dbl>    <dbl>       <dbl>       <dbl>       <dbl>
 1 Agatha          1060     1860     2000       1.06        1.86        2
 2 Angelique       2920     2940      209       2.92        2.94        0.209
 3 Annabel Lee      944     1180     1689       0.944       1.18        1.69
 4 Ardrey-A          98       98       95       0.098       0.098       0.095
 5 Ardrey          3000     2780      666       3           2.78        0.666
 6 Bellatrix-A       NA       NA       NA      NA          NA          NA
 7 Bellatrix-B       NA       NA       NA      NA          NA          NA
 8 Bellatrix-C       NA       NA       NA      NA          NA          NA
 9 Bellatrix        585     2760     2460       0.585       2.76        2.46
10 Blue Devil      1330     1820     2460       1.33        1.82        2.46
# ... with 41 more rows
```

Трансформировать несколько столбцов

`transmute` оставляет только перечисленные и новые столбцы.

```
lemurs_weights %>%
  transmute(
    name,
    across(starts_with("weight"), list(kg = ~ .x/1000), .names = "KG_{.col}"))
```

```
# A tibble: 51 x 4
   name        KG_weight_1 KG_weight_2 KG_weight_3
   <chr>             <dbl>       <dbl>       <dbl>
 1 Agatha            1.06        1.86        2
 2 Angelique         2.92        2.94        0.209
 3 Annabel Lee       0.944       1.18        1.69
 4 Ardrey-A          0.098       0.098       0.095
 5 Ardrey            3           2.78        0.666
 6 Bellatrix-A      NA          NA          NA
 7 Bellatrix-B      NA          NA          NA
 8 Bellatrix-C      NA          NA          NA
 9 Bellatrix         0.585       2.76        2.46
10 Blue Devil        1.33        1.82        2.46
# ... with 41 more rows
```

`summarise` по нескольким столбцам с группировкой

```
lemurs
```

```
# A tibble: 51 x 6
   name           sex   weight_1 weight_2 weight_3 birth_type
   <chr>          <chr>    <dbl>    <dbl>    <dbl> <chr>
 1 Nosferatu      M         2860     2505     2930 wild
 2 Poe            M         2700     2610     2680 wild
 3 Samantha       F         2242     2360     2415 wild
 4 Annabel Lee    F          944     1180     1689 captive
 5 Mephistopheles M         2760     2520     2620 wild
 6 Endora         F         2600     2360     2645 wild
 7 Ozma           F         2500     2440     2620 wild
 8 Morticia       F         2700     2550     2255 wild
 9 Blue Devil     M         1330     1820     2460 captive
10 Goblin         M         1180     1460     1150 captive
# ... with 41 more rows
```

`summarise` по нескольким столбцам с группировкой

```
lemurs %>%
  group_by(sex, birth_type) %>%
  summarise(across(starts_with("weight"), mean, na.rm = TRUE))
```

```
# A tibble: 5 x 5
# Groups:   sex [3]
  sex   birth_type weight_1 weight_2 weight_3
  <chr> <chr>         <dbl>    <dbl>    <dbl>
1 F     captive       1604.    1490.    1722.
2 F     wild          2510.    2428.    2484.
3 M     captive       1893.    1589.    1410.
4 M     wild          2773.    2545     2743.
5 <NA>  captive         NaN      NaN      NaN
```

`summarise` по нескольким столбцам с группировкой

```
lemurs %>%
  drop_na(sex) %>%
  group_by(sex, birth_type) %>%
  summarise(across(starts_with("weight"), ~ mean(.x, na.rm = TRUE)))
```

```
# A tibble: 4 x 5
# Groups:   sex [2]
  sex   birth_type weight_1 weight_2 weight_3
  <chr> <chr>         <dbl>    <dbl>    <dbl>
1 F     captive       1604.    1490.    1722.
2 F     wild          2510.    2428.    2484.
3 M     captive       1893.    1589.    1410.
4 M     wild          2773.    2545     2743.
```

`summarise` по нескольким столбцам с группировкой...

```
lemurs %>%
  drop_na(sex) %>%
  group_by(sex, birth_type) %>%
  summarise(across(starts_with("weight"), mean, na.rm = TRUE)) %>%
  ungroup() %>%
  rowwise() %>%
  mutate(avg_weight = mean(c_across(starts_with("weight")), na.rm = TRUE))
```

```
# A tibble: 4 x 6
# Rowwise:
  sex   birth_type weight_1 weight_2 weight_3 avg_weight
  <chr> <chr>         <dbl>    <dbl>    <dbl>      <dbl>
1 F     captive       1604.    1490.    1722.      1605.
2 F     wild          2510.    2428.    2484.      2474.
3 M     captive       1893.    1589.    1410.      1631.
4 M     wild          2773.    2545     2743.      2687.
```

Подсчет наблюдений

- `tally()` - количество набюдений (строк) всего
- `add_tally()` - добавляется отдельный столбец с общим количеством наблюдений

```
lemurs %>% tally()
```

```
# A tibble: 1 x 1
      n
  <int>
1    51
```

```
lemurs %>%
  select(1:2) %>%
  add_tally()
```

```
# A tibble: 51 x 3
   name           sex       n
   <chr>          <chr> <int>
 1 Nosferatu      M        51
 2 Poe            M        51
 3 Samantha       F        51
 4 Annabel Lee    F        51
 5 Mephistopheles M        51
 6 Endora         F        51
 7 Ozma           F        51
 8 Morticia       F        51
 9 Blue Devil     M        51
10 Goblin         M        51
# ... with 41 more rows
```

Подсчет наблюдений в группах

- `count()` - количество набюдений (строк) в группе
- `add_count()` - добавляется отдельный столбец с количеством наблюдений в группе

```
lemurs %>% count(sex)
```

```
# A tibble: 3 x 2
  sex       n
  <chr> <int>
1 F        26
2 M        24
3 <NA>      1
```

```
lemurs %>%
  select(1:2) %>%
  add_count(sex)
```

```
# A tibble: 51 x 3
   name           sex       n
   <chr>          <chr> <int>
 1 Nosferatu      M        24
 2 Poe            M        24
 3 Samantha       F        26
 4 Annabel Lee    F        26
 5 Mephistopheles M        24
 6 Endora         F        26
 7 Ozma           F        26
 8 Morticia       F        26
 9 Blue Devil     M        24
10 Goblin         M        24
# ... with 41 more rows
```

Фильтрация данных

- `between()`

```
lemurs %>%
  filter(between(weight_1, 900, 1100))
```

- `near()`

```
lemurs %>%
  # от 900 до 1100
  filter(near(weight_1, 1000, tol = 100))
```

```
# A tibble: 2 x 6
  name        sex   weight_1 weight_2 weight_3 birth_type
  <chr>       <chr>    <dbl>    <dbl>    <dbl> <chr>
1 Annabel Lee F          944     1180     1689 captive
2 Agatha      F         1060     1860     2000 captive
```

Фильтрация данных

- `near()`

```
lemurs %>%
  filter(near(weight_1,
              mean(weight_1, na.rm = TRUE),
              tol = sd(weight_1, na.rm = TRUE)))
```

```
# A tibble: 24 x 6
   name            sex   weight_1 weight_2 weight_3 birth_type
   <chr>           <chr>    <dbl>    <dbl>    <dbl> <chr>
 1 Poe             M         2700     2610     2680 wild
 2 Samantha        F         2242     2360     2415 wild
 3 Annabel Lee     F          944     1180     1689 captive
 4 Mephistopheles M          2760     2520     2620 wild
 5 Endora          F         2600     2360     2645 wild
 6 Ozma            F         2500     2440     2620 wild
 7 Morticia        F         2700     2550     2255 wild
 8 Blue Devil      M         1330     1820     2460 captive
 9 Goblin          M         1180     1460     1150 captive
10 Cruella         F         2050     1350     2340 captive
# ... with 14 more rows
```

Фильтрация по нескольким столбцам

```
lemurs %>%
  filter(across(starts_with("weight"), ~ . > 2500))
```

```
# A tibble: 3 x 6
  name           sex   weight_1 weight_2 weight_3 birth_type
  <chr>          <chr>    <dbl>    <dbl>    <dbl> <chr>
1 Nosferatu      M         2860     2505     2930 wild
2 Poe            M         2700     2610     2680 wild
3 Mephistopheles M         2760     2520     2620 wild
```

Фильтрация по нескольким столбцам

`if_any` оставляет те строки, где хотя бы в одном из указанных столбцов условие выполняется.

```
lemurs_weights %>%
  filter(if_any(starts_with("weight"),
   ~ . > 3000))
```

```
# A tibble: 1 x 4
  name     weight_1 weight_2 weight_3
  <chr>       <dbl>    <dbl>    <dbl>
1 Lucrezia     2560      213     3070
```

`if_all` оставляет те строки, где во всех указанных столбцах условие выполняется.

```
lemurs_weights %>%
  filter(if_all(starts_with("weight"),
   ~ . > 2500))
```

```
# A tibble: 3 x 4
  name           weight_1 weight_2
   weight_3
  <chr>             <dbl>    <dbl>
   <dbl>
1 Mephistopheles     2760     2520
   2620
2 Nosferatu          2860     2505
   2930
3 Poe                2700     2610
   2680
```

Фильтрация по нескольким столбцам

Фильтрация пропущенных значений - все значения должны быть не `NA`.

```
lemurs_weights %>%
  drop_na(where(is.numeric))
```

```
# A tibble: 34 x 4
   name        weight_1 weight_2 weight_3
   <chr>          <dbl>    <dbl>    <dbl>
 1 Agatha          1060     1860     2000
 2 Angelique       2920     2940      209
 3 Annabel Lee      944     1180     1689
 4 Ardrey-A          98       98       95
 5 Ardrey          3000     2780      666
 6 Bellatrix        585     2760     2460
 7 Blue Devil      1330     1820     2460
 8 Caliban         2080      296.     641
 9 Claudia          740     2550     2390
10 Cruella         2050     1350     2340
# ... with 24 more rows
```

```
lemurs_weights %>%
  filter(if_all(where(is.numeric), ~
    !is.na(.x)))
```

```
# A tibble: 34 x 4
   name        weight_1 weight_2 weight_3
   <chr>          <dbl>    <dbl>    <dbl>
 1 Agatha          1060     1860     2000
 2 Angelique       2920     2940      209
 3 Annabel Lee      944     1180     1689
 4 Ardrey-A          98       98       95
 5 Ardrey          3000     2780      666
 6 Bellatrix        585     2760     2460
 7 Blue Devil      1330     1820     2460
 8 Caliban         2080      296.     641
 9 Claudia          740     2550     2390
10 Cruella         2050     1350     2340
# ... with 24 more rows
```

Фильтрация по нескольким столбцам

Фильтрация пропущенных значений - хоть одно значение не `NA`.

```
lemurs_weights %>%
  filter(if_any(where(is.numeric), ~ !is.na(.x)))
```

```
# A tibble: 37 x 4
   name        weight_1 weight_2 weight_3
   <chr>          <dbl>    <dbl>    <dbl>
 1 Agatha          1060     1860     2000
 2 Angelique       2920     2940      209
 3 Annabel Lee      944     1180     1689
 4 Ardrey-A          98       98       95
 5 Ardrey          3000     2780      666
 6 Bellatrix        585     2760     2460
 7 Blue Devil      1330     1820     2460
 8 Caliban         2080      296.     641
 9 Claudia          740     2550     2390
10 Cruella         2050     1350     2340
# ... with 27 more rows
```

Что почитать про продвинутый dplyr

- dplyr cheatsheet
- Data Wrangling by Suzan Baert
- dplyr - Column-wise operations
- dplyr - Row-wise operations
- `?across` и прочие хелпы...