Data wrangling with dplyr

Анастасия Миллер, июль 2018

Хорошие данные

Pipe operator %>%

Базовое использование:

%>% подставляет то, что у него слева, первым аргументом в вызов справа:

- `x %>% f` – то же, что и `f(x)`
- `x %>% f(y)` – то же, что и `f(x, y)`
- `x %>% f %>% g %>% h` – то же, что и `h(g(f(x)))`

Pipe operator %>%

С указанием места:

- `x %>% f(y, .)` – то же, что и `f(y, x)`
- `x %>% f(y, z = .)` – то же, что и `f(y, z = x)`

Вопросы к данным

- Сколько всего фанфиков в нашем наборе?

Вопросы к данным

- Сколько всего фанфиков в нашем наборе?

```
df %>% nrow()
```

```
[1] 14424
```

Фильтрация: `filter` и `select`

- `filter` позволяет выбрать ряды, соответствующие некоторому условию (булевой маске):

```
df %>% filter(size_cat == 'small') %>% nrow()
## [1] 11362
```

- `select` позволяет выбрать колонки:

```
df %>% select(id, starts_with('size'))
           id size_cat size_kb
1      119387    small      22
2      119330    small       3
3      109013    large     612
…           …        …       …
```

Вопросы к данным

- Сколько фанфиков имеют размер больше 100Кб
- Когда опубликован самый первый фанфик? И что это был за фанфик?

Вопросы к данным

- Сколько фанфиков имеют размер больше 100Кб?

```
df %>% filter(size_kb > 100) %>% nrow()
```

```
[1] 1918
```

- Когда опубликован самый первый фанфик? И что это был за фанфик?

```
df %>% select(id, title, published)
   %>% filter(published == min(published))
```

```
# A tibble: 25 x 3
      id title                                                  published
   <int> <chr>                                                  <date>
 1     2 Ни что не меняется так часто, как прошлое...           2005-03-30
 2    31 То, что думаю                                          2005-03-30
 3    32 Фантазия                                               2005-03-30
 …     … …                                                      …
```

Разбор решений

Когда опубликован самый первый фанфик?
И что это был за фанфик?

- Через фильтр:
  ```
  df %>% filter(published == min(published))
  ```

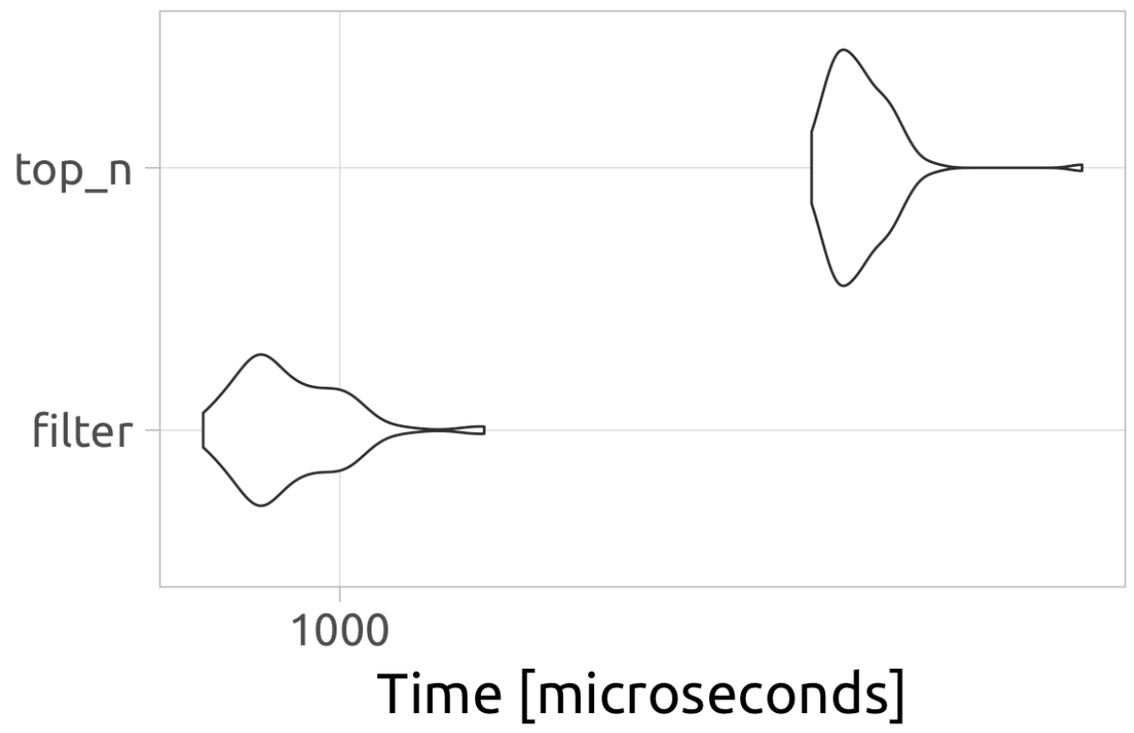
- Через `top_n`:
  ```
  df %>% top_n(1, published)
  ```

Какое из решений быстрее?

Разбор решений

Какое из решений быстрее?

```
microbenchmark(
  'filter' = {df %>% filter(published == max(published))},
  'top_n' = {df %>% top_n(1, published)}
) %>% autoplot()
```



Изменение данных: `mutate` и `arrange`

`mutate` добавляет новые переменные в набор, сохраняя старые:

```
df %>% select(id, title, translated)
    %>% mutate(
      orig_lang = ifelse(translated, 'Unknown', 'Russian')
)
# A tibble: 14,424 x 4
       id title                                translated orig_lang
    <int> <chr>                                <lgl>      <chr>
 1 119387 Одно ветреное утро                   TRUE       Unknown
 2 119330 Локальный апокалипсис в Хайленде     FALSE      Russian
 3 109013 Мысли и возможности                  FALSE      Russian
 …      … …                                    …          …
```

Изменение данных: `mutate` и `arrange`

`transmute` добавляет новые переменные в набор, удаляя старые:

```
df %>% select(id, title, translated)
    %>% transmute(
        orig_lang = ifelse(translated, 'Unknown', 'Russian')
)
# A tibble: 14,424 x 1
   orig_lang
   <chr>
 1 Unknown
 2 Russian
 3 Russian
 4 Russian
```

Изменение данных: `mutate` и `arrange`

`arrange` сортирует по переменной:

```
df %>% select(title, size_kb, published)
    %>% arrange(size_kb, published)
# A tibble: 14,424 x 3
  title                                size_kb published
  <chr>                                  <int> <date>
1 Все делают это                             0 2011-07-15
2 Гусеница                                   0 2014-05-02
3 Без названия.                              1 2005-03-30
```

Изменение данных: `mutate` и `arrange`

`desc` меняет направление сортировки:

```
df %>% select(title, size_kb, published)
    %>% arrange(published, desc(size_kb))
# A tibble: 14,424 x 3
   title                                      size_kb published
   <chr>                                        <int> <date>
 1 Гарри Поттер и Наследники Слизерина           1457 2005-03-30
 2 Гарри Поттер и Обряд Защиты Рода               477 2005-03-30
 3 Гостья из Шармбатона                           247 2005-03-30
```

Вопросы к данным

- Какой фанфик писался дольше всего? Сколько времени заняло написание?

- Из тех фанфиков, которые писались больше года, какой был раньше всех начат и когда это случилось?

- Когда опубликовали первый переводной фанфик?

- Сколько времени прошло от открытия сайта до завершения пятого опубликованного перевода?

Полезно знать:

```
df$published %>% class()
[1] "character"


ymd(df$published) %>% class()
[1] "Date"
```

Вопросы к данным

- Какой фанфик писался дольше всего? Сколько времени заняло написание?

```
df %>% mutate(
  published = ymd(published),
  last_update = ymd(last_update),
  time_to_complete = last_update - published
) %>% filter(
  time_to_complete == max(time_to_complete)
)
```

```
  id     title   published last_update size_kb time_to_complete
1 925 Наследник  2005-04-06  2018-02-23    2798        4706 days
```

Вопросы к данным

- Из тех фанфиков, которые писались больше года, какой был раньше всех начат и когда это случилось?

```
df %>% mutate(
  published = ymd(published),
  last_update = ymd(last_update),
  time_to_complete = last_update - published
) %>% filter(
  time_to_complete > years(1)
) %>% filter(
  published == min(published)
)
```

```
  id                 title  published last_update time_to_complete
1 679          Калейдоскоп  2005-04-04  2006-12-24          629 days
2  52 Превратности Судьбы  2005-04-04  2006-11-26          601 days
```

Вопросы к данным

- Когда опубликовали первый переводной фанфик?

```
df %>% filter(translated)
    %>% pull(published) %>% min()
```

```
[1] "2005-03-30"
```

- Сколько времени прошло от открытия сайта до завершения пятого опубликованного перевода?

```
opening_day <- ymd(min(df$published))
fifth_translation_complete <- df %>%
  filter(translated) %>% arrange(published) %>%
  slice(5) %>% pull(last_update) %>% ymd()
fifth_translation_complete - opening_day
```

```
Time difference of 240 days
```

Агрегация: `group_by` и `summarise`

`summarise` считает статистики:

```
df %>% summarise(
  mean_size = mean(size_kb),
  mean_ttc = mean(ymd(last_update) - ymd(published))
)
```

```
  mean_size       mean_ttc
1  69.34332 42.30359 days
```

Агрегация: `group_by` и `summarise`

`group_by` позволяет считать внутри группы, а не во всём наборе данных:

```
df %>% group_by(
  size_cat
) %>% summarise(
  mean_size = mean(size_kb),
  mean_ttc = mean(ymd(last_update) - ymd(published)),
  median_ttc = median(ymd(last_update) - ymd(published))
)
  size_cat mean_size         mean_ttc median_ttc
1    large  588.1886 347.928778 days   113 days
2   medium  106.2264  79.485418 days     4 days
3    small   15.3304   7.735434 days     0 days
```

Вопросы к данным

- Здесь есть есть авторские произведения и переводы. Сколько авторских, сколько переводов?

- Как часто появляются новые фанфики? Посчитайте, сколько в среднем фанфиков появляется в месяц.

- Отличается ли частота публикаций в месяц, рассчитанная за всё время существования сайта, от рассчитанной за последние пять лет?

- Отличается ли частота публикаций для фанфиков разных рейтингов?

Разбор решений

Здесь есть есть авторские произведения и переводы. Сколько авторских, сколько переводов?

- Через `group_by`:
```
df %>% group_by(translated) %>% count()
```

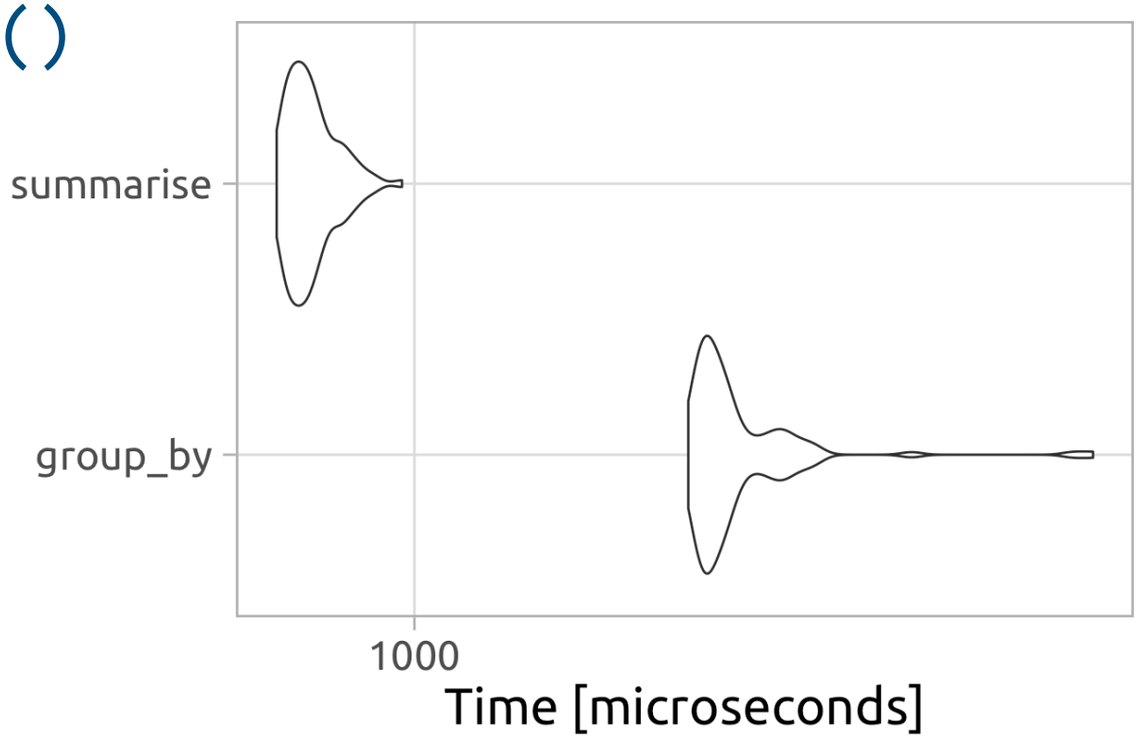
- Через `summarise`:
```
df %>% summarise(
  authored = sum(!translated),
  translated = sum(translated)
)
```

Какое из решений быстрее?

Разбор решений

Какое из решений быстрее?

```
microbenchmark(
   'group_by' = {df %>% group_by(translated) %>% count()},
   'summarise' = {df %>% summarise(
      authored = sum(!translated),
      translated = sum(translated))}
) %>% autoplot()
```



Вопросы к данным

- Как часто появляются новые фанфики? Посчитайте, сколько в среднем фанфиков появляется в месяц.

```
df %>% group_by(
  year = year(ymd(published)),
  month = month(ymd(published))
) %>% count() %>% ungroup() %>% summarise(
  monthly_rate_total = mean(n)
)
```

```
  monthly_rate_total
1           93.05806
```

Вопросы к данным

- Отличается ли частота публикаций в месяц, рассчитанная за всё время существования сайта, от рассчитанной за последние пять лет?

```
df %>% group_by(
  year = year(ymd(published)),
  month = month(ymd(published))
) %>% count() %>% ungroup() %>% summarise(
  monthly_rate_total = mean(n),
  monthly_rate_last_5 = mean(
    n[year >= year(today()) - 5])
)
```

```
  monthly_rate_total monthly_rate_last_5
1           93.05806             149.7164
```

Вопросы к данным

- Отличается ли частота публикаций для фанфиков разных рейтингов?

```
df %>% group_by(
  rating,
  year = year(ymd(published)),
  month = month(ymd(published))
) %>% count() %>% group_by(rating) %>%
  summarise(freq = mean(n)) %>% arrange(freq)
```

```
     rating     freq
1 не указан 14.88889
2     NC-17 15.78169
3         R 19.78912
4   General 27.94928
5     PG-13 32.97973
```

Бонусное задание

- Отличается ли частота публикаций для фанфиков разных рейтингов?

- **Является ли различие статистически значимым?**

Плохие данные

Работа со списками: nest и unnest

- unnest разворачивает список, создавая для каждого элемента новый ряд в данных:

```
df %>% select(id, title, genre)
   %>% slice(c(1,3,5))
   %>% as.data.frame()
```

```
     id                    title                          genre
1 119387  Одно ветреное утро                Romance, Angst
2 109013 Мысли и возможности  Drama, Hurt-Comfort, POV
3 119132      Найдите Пэкки!                         Romance
```

Работа со списками: nest и unnest

- unnest разворачивает список, создавая для каждого элемента новый ряд в данных:

```
df %>% select(id, title, genre)
   %>% slice(c(1,3,5)) %>% unnest(genre)
   %>% as.data.frame()
```

```
     id                    title        genre
1 119387  Одно ветреное утро         Romance
2 119387  Одно ветреное утро           Angst
3 109013 Мысли и возможности           Drama
4 109013 Мысли и возможности  Hurt-Comfort
5 109013 Мысли и возможности             POV
6 119132      Найдите Пэкки!         Romance
```

Работа со списками: nest и unnest

- nest сворачивает все значения в список:

```
df %>% select(published, authors)
   %>% slice(1:10)
```

```
   published       authors
1  2018-07-21          NULL
2  2018-07-18 98623, 101967
3  2017-10-12        398169
4  2018-07-16        418727
5  2018-07-17        418727
6  2018-07-16        418727
7  2018-07-16        220954
8  2018-07-18        418727
9  2017-08-03        192961
10 2018-07-18        218337
```

Работа со списками: nest и unnest

- nest сворачивает все значения в список:

```
df %>% select(published, authors)
    %>% slice(1:10) %>% nest(authors)
```

```
    published                          data
1  2018-07-21                          NULL
2  2018-07-18 98623, 101967, 418727, 218337
3  2017-10-12                        398169
4  2018-07-16        418727, 418727, 220954
5  2018-07-17                        418727
6  2017-08-03                        192961
```

Вопросы к данным

- Каков самый популярный жанр?
- Каков был самый популярный жанр в каждом из последних восьми лет?
- С какими жанрами чаще всего сочетается AU (alternative universe)? сложно

Вопросы к данным

- Каков самый популярный жанр?

```
df %>%
  unnest(genre) %>%
  select(genre) %>%
  group_by(genre) %>%
  count() %>% arrange(desc(n))
```

```
            genre    n
1         Romance 5313
2              AU 4380
3           Drama 4189
4           Humor 3074
```

Вопросы к данным

- Каков был самый популярный жанр в каждом из последних пяти лет?

```
df %>% unnest(genre) %>% select(published, genre) %>%
  count(year = year(published), genre) %>%
  filter(year >= year(today()) - 5) %>%
  group_by(year) %>% filter(n == max(n)) %>%
  arrange(desc(year))
```

```
  year   genre   n
1 2018      AU 324
2 2017      AU 642
3 2016      AU 606
4 2015 Romance 597
5 2014      AU 615
6 2013 Romance 712
```

Вопросы к данным

- С какими жанрами чаще всего сочетается AU (alternative universe)?

```
df %>% filter(
  (genre %>% sapply(length) >= 2) &
    (sapply(genre, function(x) 'AU' %in% x))
) %>% unnest(genre) %>% filter(
  genre != 'AU'
) %>% count(genre) %>% arrange(desc(n))
```

```
           genre    n
1        Romance 1473
2          Drama 1153
3          Humor  960
4          Angst  751
```

Работа с untidy data: gather и spread

- gather вытягивает данные:

```
df %>% select(id, published, last_update)
    %>% slice(c(1,3,5))
    %>% as.data.frame()
```

```
      id                    title  published last_update
1 119387  Одно ветреное утро  2018-07-21  2018-07-21
2 109013 Мысли и возможности  2017-10-12  2018-07-16
3 119132      Найдите Пэкки!  2018-07-17  2018-07-17
```

Работа с untidy data: gather и spread

- gather вытягивает данные:

```
df %>% select(id, published, last_update)
   %>% slice(c(1,3,5))
   %>% gather("action", "date", published, last_update)
   %>% as.data.frame()
```

```
     id                  title      action       date
1 119387  Одно ветреное утро    published 2018-07-21
2 109013 Мысли и возможности    published 2017-10-12
3 119132      Найдите Пэкки!    published 2018-07-17
4 119387  Одно ветреное утро  last_update 2018-07-21
5 109013 Мысли и возможности  last_update 2018-07-16
6 119132      Найдите Пэкки!  last_update 2018-07-17
```

Работа с untidy data: gather и spread

- `spread` собирает данные в широкую таблицу:

```
df %>% count(rating, year = year(published))
```

```
      rating year   n
1  не указан 2006   6
2  не указан 2007   2
3  не указан 2008   2
4  не указан 2009   8
5  не указан 2010 163
6  не указан 2011 354
…             …    …    …
```

Работа с untidy data: gather и spread

- spread собирает данные в широкую таблицу:

```
df %>% count(rating, year = year(published))
    %>% spread(year, n)
```

	rating	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018
1	не указан	NA	6	2	2	8	163	354	1	NA	NA	NA	NA	NA	NA
2	General	23	48	15	11	17	162	355	389	464	358	496	542	718	259
3	PG-13	98	82	50	22	40	295	433	407	482	476	678	767	733	318
4	R	43	45	32	10	29	162	225	238	316	381	398	420	405	205
5	NC-17	41	57	26	6	20	107	151	218	333	302	260	300	286	134

Вопросы к данным

- Соберите отчёт по количеству публикаций в пяти наиболее популярных жанрах за последние 10 лет. Каждый год — отдельная колонка.

Вопросы к данным

- Соберите отчёт по количеству публикаций в пяти наиболее популярных жанрах за последние 10 лет. Каждый год — отдельная колонка.

```
top_5_genres <- df %>%
  filter(year(published) >= year(today()) - 10) %>%
  unnest(genre) %>% count(genre) %>%
  top_n(5, n) %>% pull(genre)

df %>%
  filter(year(published) >= year(today()) - 10) %>%
  unnest(genre) %>% filter(genre %in% top_5_genres) %>%
  count(genre, year = year(published)) %>%
  spread(year, n)
```