RUDN university

5100

Программирование в среде R

Шевцов Василий Викторович,
директор ДИТ РУДН, shevtsov_vv@rudn.university

Операции с данными

Нормализация

- **первая нормальная форма (1NF);**
- **вторая нормальная форма (2NF);**
- **третья нормальная форма (3NF);**
- **нормальная форма Бойса-Кодда (BCNF);**
- **четвертая нормальная форма (4NF);**
- **пятая нормальная форма, или нормальная форма проекции-соединения (5NF или PJ/NF).**
- **Доменно-ключевая нормальная форма (DKNF)**
- **Шестая нормальная форма (6NF)**
- **Основные свойства нормальных форм:**
  - каждая следующая нормальная форма в некотором смысле лучше предыдущей;
  - при переходе к следующей нормальной форме свойства предыдущих нормальных свойств сохраняются.

Первая нормальная форма 1NF

Таблица нормализована (эквивалентно — находится в первой нормальной форме) тогда и только тогда, когда она является прямым и верным представлением некоторого отношения. Конкретнее, рассматриваемая таблица должна удовлетворять следующим пяти условиям:

- Нет упорядочивания строк сверху-вниз (другими словами, порядок строк не несет в себе никакой информации).
- Нет упорядочивания столбцов слева-направо (другими словами, порядок столбцов не несет в себе никакой информации).
- Нет повторяющихся строк.
- Каждое пересечение строки и столбца содержит ровно одно значение из соответствующего домена (и больше ничего).
- Все столбцы являются обычными

Первая нормальная форма 1NF

Сотрудник	Номер телефона
Иванов И. И.	283-56-82 390-57-34
Петров П. П.	708-62-34

Сотрудник	Номер телефона
Иванов И. И.	283-56-82
Иванов И. И.	390-57-34
Петров П. П.	708-62-34

Вторая нормальная форма 2NF

- Находится во второй нормальной форме (2NF) в том и только в том случае, когда находится в 1NF, и каждый неключевой атрибут полностью зависит от первичного ключа.
- Т.е. таблицы содержат данные только об одном объекте и этот объект идентифицирован первичным ключом

Третья нормальная форма 3NF

- Находится во второй нормальной форме (2NF)и все неключевые столбцы взаимно независимы
- Например столбец, значения которого получены в результате вычислений других столбцов, является зависимым

tidyverse

Загрузка и трансформация данных

tidyverse

tidyverse — это набор пакетов:

- ggplot2, для визуализации
- tibble, для работы с тибблами, современный вариант датафрейма
- tidyr, для формата tidy data
- readr, для чтения файлов в R
- purrr, для функционального программирования
- dplyr, для преобразованиия данных
- stringr, для работы со строковыми переменными
- forcats, для работы с переменными-факторами

Полезно также знать о следующих:

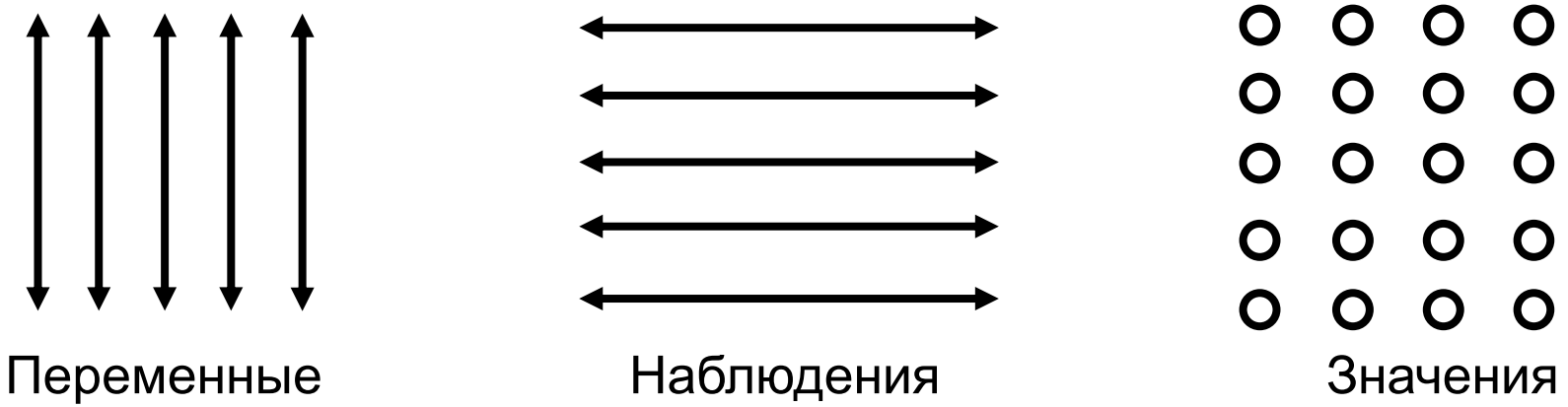
- readxl, для чтения .xls и .xlsx
- read_tsv() – для файлов с табуляцией в качестве разделителя
- read_csv2() – для файлов с точкой с запятой в качестве разделителя
- read_delim(file = "...", delim = "...") – для файлов с любым разделителем, задаваемым аргументом delim
- jsonlite, для работы с JSON
- rvest, для веб-скреппинга
- lubridate, для работы с временем
- tidytext, для работы с текстами
- broom, для перевода в tidy формат статистические модели

tidyr

Концепция TidyData. Требования к данным

Цель tidyr — помочь привести данные к так называемому аккуратному виду. Аккуратные данные — это данные, где:

- Каждая переменна должна иметь собственный столбец
- Каждое наблюдение имеет собственную строку
- Каждое значение имеет собственную ячейку



Все три правила взаимосвязаны.
Соблюдение только двух невозможно.

Основные функции входящие в пакет tidyr

tidyr содержит набор функции предназначенных для трансформации таблиц:

- fill() — заполнение пропущенных значений в столбце, предыдущими значениями;
- separate() — разбивает одно поле на несколько через разделитель;
- unite() — совершает операцию объединения нескольких полей в одно, действие обратное функции separate();
- pivot_longer() — функция, преобразующая данные из широкого формата в длинный;
- pivot_wider() — функция, преобразующая данные из длинного формата в широкий. Операция обратная той, которую осуществляет функция pivot_longer().
- gather() **устаревшая** — функция, преобразующая данные из широкого формата в длинный;
- spread() **устаревшая** — функция, преобразующая данные из длинного формата в широкий. Операция обратная той, которую осуществляет функция gather().

pivot_longer()

Аргументы функции pivot_longer()

- cols, описывает, какие столбцы необходимо объединить
- names_to дает имя переменной, которая будет создана из имён столбцов, которые объединяются
- values_to дает имя переменной, которая будет создана из данных, хранящихся в значениях ячеек объединённых столбцов

Набор данных, требующий преобразования

Набор данных who, который предоставляется вместе с пакетом tidyr. Этот набор данных содержит информацию предоставляемую международной организацией здравоохранения о заболеваемости туберкулёзом.

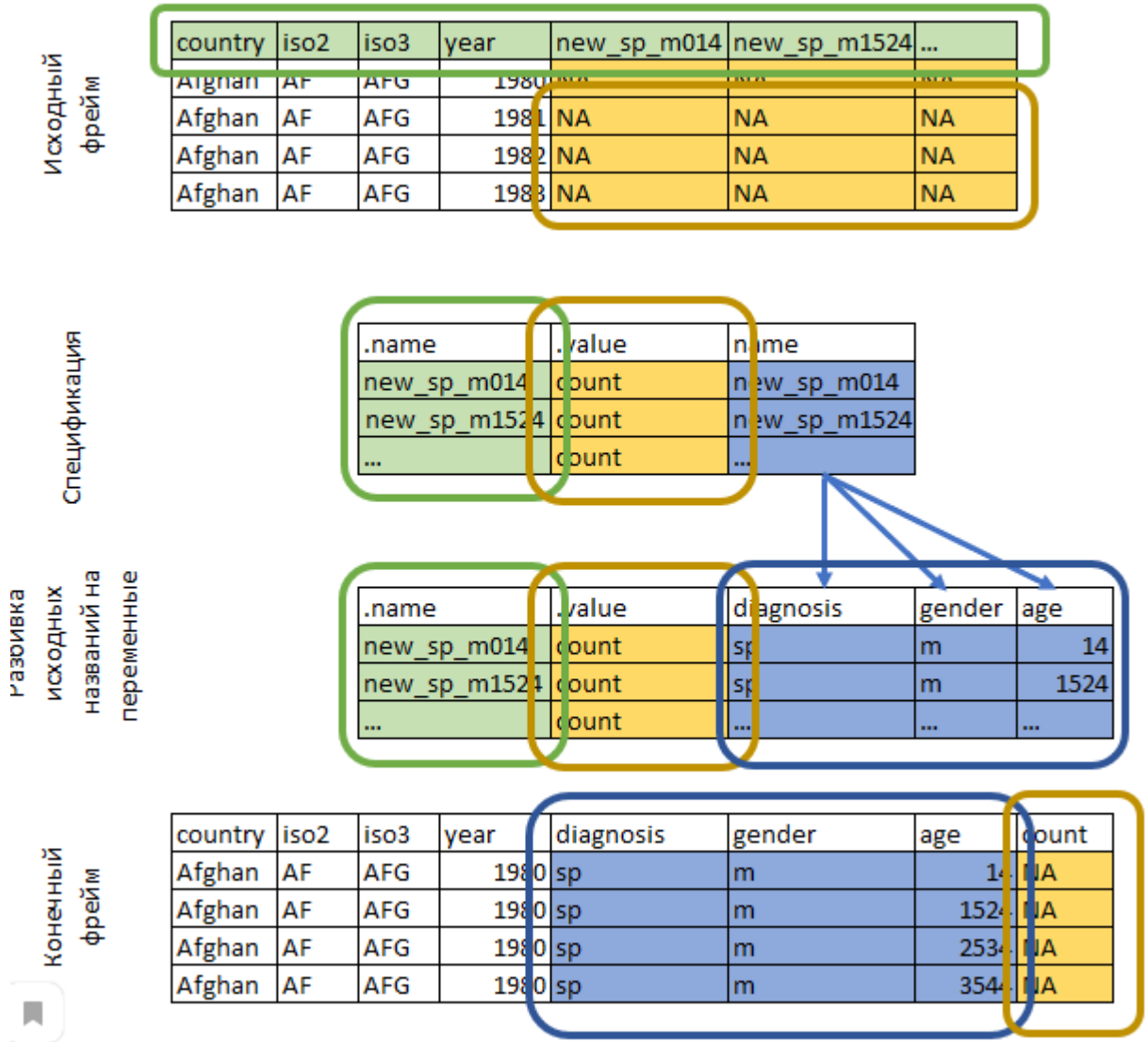
```
> tidyr::who
# A tibble: 7,240 x 60
   country iso2  iso3   year new_sp_m014 new_sp_m1524 new_sp_m2534 new_sp_m3544 new_sp_m4554 new_sp_m5564 new_sp_m65 new_sp_f014
   <chr>   <chr> <chr> <int>       <int>        <int>        <int>        <int>        <int>        <int>      <int>       <int>
 1 Afghan~ AF    AFG    1980          NA           NA           NA           NA           NA           NA         NA          NA
 2 Afghan~ AF    AFG    1981          NA           NA           NA           NA           NA           NA         NA          NA
 3 Afghan~ AF    AFG    1982          NA           NA           NA           NA           NA           NA         NA          NA
 4 Afghan~ AF    AFG    1983          NA           NA           NA           NA           NA           NA         NA          NA
 5 Afghan~ AF    AFG    1984          NA           NA           NA           NA           NA           NA         NA          NA
 6 Afghan~ AF    AFG    1985          NA           NA           NA           NA           NA           NA         NA          NA
 7 Afghan~ AF    AFG    1986          NA           NA           NA           NA           NA           NA         NA          NA
 8 Afghan~ AF    AFG    1987          NA           NA           NA           NA           NA           NA         NA          NA
 9 Afghan~ AF    AFG    1988          NA           NA           NA           NA           NA           NA         NA          NA
10 Afghan~ AF    AFG    1989          NA           NA           NA           NA           NA           NA         NA          NA
# ... with 7,230 more rows, and 48 more variables: new_sp_f1524 <int>, new_sp_f2534 <int>, new_sp_f3544 <int>, new_sp_f4554 <int>,
#   new_sp_f5564 <int>, new_sp_f65 <int>, new_sn_m014 <int>, new_sn_m1524 <int>, new_sn_m2534 <int>, new_sn_m3544 <int>,
#   new_sn_m4554 <int>, new_sn_m5564 <int>, new_sn_m65 <int>, new_sn_f014 <int>, new_sn_f1524 <int>, new_sn_f2534 <int>,
#   new_sn_f3544 <int>, new_sn_f4554 <int>, new_sn_f5564 <int>, new_sn_f65 <int>, new_ep_m014 <int>, new_ep_m1524 <int>,
#   new_ep_m2534 <int>, new_ep_m3544 <int>, new_ep_m4554 <int>, new_ep_m5564 <int>, new_ep_m65 <int>, new_ep_f014 <int>,
#   new_ep_f1524 <int>, new_ep_f2534 <int>, new_ep_f3544 <int>, new_ep_f4554 <int>, new_ep_f5564 <int>, new_ep_f65 <int>,
#   newrel_m014 <int>, newrel_m1524 <int>, newrel_m2534 <int>, newrel_m3544 <int>, newrel_m4554 <int>, newrel_m5564 <int>,
#   newrel_m65 <int>, newrel_f014 <int>, newrel_f1524 <int>, newrel_f2534 <int>, newrel_f3544 <int>, newrel_f4554 <int>,
#   newrel_f5564 <int>, newrel_f65 <int>
```

Набор данных, требующий преобразования

В названиях этих столбцов хранится следующая информация:

- Префикс new_ говорит о том, что столбец содержит данные о новых случаях заболевания туберкулёом, текущий дата фрейм содержит информацию только по новым заболеваниям, поэтому данный префикс в текущем контексте не несёт никакой смысловой нагрузки.
- sp/rel/sp/ep описывает способ диагностики заболевания.
- m/f пол пациента.
- 014/1524/2535/3544/4554/65 возрастной диапазон пациента.

Схема преобразований



RUDN university

tibble

tibble

Пакет tibble – является альтернативой штатного датафрейма в R. Существует встроенная переменная month.name:

```
> month.name
 [1] "January"   "February"  "March"     "April"     "May"       "June"      "July"      "August"
"September" "October"   "November"  "December"

> data.frame(id = 1:12,
+            months = month.name,
+            n_letters = nchar(months))
Error in nchar(months) :
  cannot coerce type 'closure' to vector of type 'character'

> data.frame(id = 1:12,
+            months = month.name,
+            n_letters = nchar(month.name))

   id    months n_letters
1   1  January         7
2   2  February        8
3   3  March           5
4   4  April           5
5   5  May             3
```

tibble

Одно из отличий tibble от базового датафрейма – возможность использовать создаваемые “по ходу пьесы переменные”

```
> tibble(id = 1:12,
+        months = month.name,
+        n_letters = nchar(months))

# A tibble: 12 x 3
      id months    n_letters
   <int> <chr>         <int>
 1     1 January         7
 2     2 February        8
 3     3 March           5
 4     4 April           5
 5     5 May             3
 6     6 June            4
 7     7 July            4
```

tibble

Если в окружении пользователя уже есть переменная с датафреймом, его легко можно переделать в tibble при помощи функции as_tibble():

as_tibble(df)

Функицонально tibble от data.frame ничем не отличается, однако существует ряд несущественных отличий. Кроме того стоит помнить, что многие функции из tidyverse возвращают именно tibble, а не data.frame.

tibble

Различия также касаются в операциях вывода на печать и извлечении поднаборов.

- выводится top 10
- число столбцов в пределах экрана
- в дополнение к имени выводится тип данных
- print(df, n=число_строк, width=ширина_столбца)
  width=Inf – все столбцы
- options(tibble.print_max=n, tibble.print_min=m)
  если количество строк больше m, вывести на печать n строк
- options(tibble.print_min=Inf) – вывод всех строк
- options(tibble.print_ width=Inf) – вывод всех строк

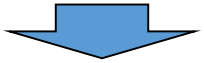
Извлечение поднаборов

- генерирует сообщений об ошибке при обращении к несуществующему столбцу

Сведение столбцов

```
install.packages("tidyr")
library(tidyr)
df1 <- read.csv2("C:\\Численность населения 2018.csv")
df2 <- gather(
        data = df1,
        "X2005","X2010","X2011","X2012","X2013","X2014","X2015","X2016","X2017",
        key = "Year",
        value = "Population")
```

```
> df1
                Страна                          Округ                      Область X2005 X2010 X2011 X2012 X2013 X2014 X2015 X2016 X2017 X2018
1  Российская Федерация Центральный федеральный округ  Белгородская область  1512  1532  1536  1541  1544  1548  1550  1553  1550  1548
2  Российская Федерация Центральный федеральный округ      Брянская область  1327  1275  1264  1254  1242  1233  1226  1221  1211  1200
3  Российская Федерация Центральный федеральный округ  Владимирская область  1486  1441  1432  1422  1413  1406  1397  1390  1378  1366
4  Российская Федерация Центральный федеральный округ  Воронежская область  2361  2335  2332  2330  2329  2331  2333  2335  2333  2328
5  Российская Федерация Центральный федеральный округ   Ивановская область  1102  1060  1054  1049  1043  1037  1030  1023  1015  1004
6  Российская Федерация Центральный федеральный округ    Калужская область  1023  1009  1008  1006  1005  1011  1010  1014  1012  1009
7  Российская Федерация Центральный федеральный округ  Костромская область   700   666   662   659   656   654   651   648   643   637
8  Российская Федерация Центральный федеральный округ      Курская область  1178  1126  1122  1119  1119  1117  1120  1123  1115  1107
9  Российская Федерация Центральный федеральный округ     Липецкая область  1194  1172  1166  1162  1160  1158  1156  1156  1150  1144
10 Российская Федерация Центральный федеральный округ    Московская область  6784  7106  7199  7048  7134  7231  7319  7423  7503  7599
```



```
> df2
                Страна                          Округ                      Область  Year Population
1  Российская Федерация Центральный федеральный округ  Белгородская область X2005       1512
2  Российская Федерация Центральный федеральный округ      Брянская область X2005       1327
3  Российская Федерация Центральный федеральный округ  Владимирская область X2005       1486
4  Российская Федерация Центральный федеральный округ  Воронежская область X2005       2361
5  Российская Федерация Центральный федеральный округ   Ивановская область X2005       1102
6  Российская Федерация Центральный федеральный округ    Калужская область X2005       1023
7  Российская Федерация Центральный федеральный округ  Костромская область X2005        700
8  Российская Федерация Центральный федеральный округ      Курская область X2005       1178
9  Российская Федерация Центральный федеральный округ     Липецкая область X2005       1194
10 Российская Федерация Центральный федеральный округ    Московская область X2005       6784
```

Сведение столбцов

```
df2_2 <- pivot_longer(
            data = df1,
            cols=c("X2005","X2010","X2011","X2012","X2013","X2014",
                   "X2015","X2016","X2017","X2018"),
            names_to = "Year",
            values_to = "Population"
)
```

```
df2_2 <- pivot_longer(
            data = df1,
            cols=starts_with("X"),
            names_to = "Year",
            values_to = "Population"
)
```

```
> df2_2
# A tibble: 820 x 5
   Страна               Округ                          Область              Year  Population
   <chr>                <chr>                          <chr>                <chr>      <int>
 1 Российская Федерация Центральный федеральный округ Белгородская область X2005       1512
 2 Российская Федерация Центральный федеральный округ Белгородская область X2010       1532
 3 Российская Федерация Центральный федеральный округ Белгородская область X2011       1536
 4 Российская Федерация Центральный федеральный округ Белгородская область X2012       1541
 5 Российская Федерация Центральный федеральный округ Белгородская область X2013       1544
 6 Российская Федерация Центральный федеральный округ Белгородская область X2014       1548
 7 Российская Федерация Центральный федеральный округ Белгородская область X2015       1550
 8 Российская Федерация Центральный федеральный округ Белгородская область X2016       1553
 9 Российская Федерация Центральный федеральный округ Белгородская область X2017       1550
10 Российская Федерация Центральный федеральный округ Белгородская область X2018       1548
# ... with 810 more rows
```

pivot_longer() arguments

data
A data frame to pivot.
cols
Columns to pivot into longer format. This takes a tidyselect specification.
names_to
A string specifying the name of the column to create from the data stored in the column names of data.
Can be a character vector, creating multiple columns, if names_sep or names_pattern is provided.
names_prefix
A regular expression used to remove matching text from the start of each variable name.
names_sep, names_pattern
If names_to contains multiple values, these arguments control how the column name is broken up.
names_sep takes the same specification as separate(), and can either be a numeric vector (specifying positions to break on), or a single string (specifying a regular expression to split on).
names_pattern takes the same specification as extract(), a regular expression containing matching groups (()).
If these arguments does not give you enough control, use pivot_longer_spec() to create a spec object and process manually as needed.
names_ptypes, values_ptypes
A list of of column name-prototype pairs. A prototype (or ptype for short) is a zero-length vector (like integer() or numeric()) that defines the type, class, and attributes of a vector.
If not specified, the type of the columns generated from names_to will be character, and the type of the variables generated from values_to will be the common type of the input columns used to generate them.
names_repair
What happen if the output has invalid column names? The default, "check_unique" is to error if the columns are duplicated. Use "minimal" to allow duplicates in the output, or "unique" to de-duplicated by adding numeric suffixes. See vctrs::vec_as_names() for more options.
values_to
A string specifying the name of the column to create from the data stored in cell values. If names_to is a character containing the special .value sentinel, this value will be ignored, and the name of the value column will be derived from part of the existing column names.
values_drop_na
If TRUE, will drop rows that contain only NAs in the value_to column. This effectively converts explicit missing values to implicit missing values, and should generally be used only when missing values in data were created by its structure.

Рассредоточение столбцов

df3 <- spread(data = df2, key = "Year", value = "Population")

```
> df2
                Страна                           Округ                       Область  Year Population
1  Российская федерация  Центральный федеральный округ         Белгородская область X2005       1512
2  Российская федерация  Центральный федеральный округ             Брянская область X2005       1327
3  Российская федерация  Центральный федеральный округ         Владимирская область X2005       1486
4  Российская федерация  Центральный федеральный округ          Воронежская область X2005       2361
5  Российская федерация  Центральный федеральный округ           Ивановская область X2005       1102
6  Российская федерация  Центральный федеральный округ             Калужская область X2005       1023
7  Российская федерация  Центральный федеральный округ          Костромская область X2005        700
8  Российская федерация  Центральный федеральный округ               Курская область X2005       1178
9  Российская федерация  Центральный федеральный округ             Липецкая область X2005       1194
10 Российская федерация  Центральный федеральный округ           Московская область X2005       6784
```

```
> df3
                Страна                                Округ                      Область X2005 X2010 X2011 X2012 X2013 X2014 X2015 X2016 X2017 X2018
1  Российская федерация  Дальневосточный федеральный округ              Амурская область   861   829   821   817   811   810   806   802   798   794
2  Российская федерация  Дальневосточный федеральный округ  Еврейская автономная область   182   176   175   173   171   169   166   164   162   160
3  Российская федерация  Дальневосточный федеральный округ             Забайкальский край  1124  1106  1100  1095  1090  1087  1083  1079  1073  1066
4  Российская федерация  Дальневосточный федеральный округ                Камчатский край   337   322   320   320   320   317   316   315   316   315
5  Российская федерация  Дальневосточный федеральный округ            Магаданская область   170   156   155   152   150   148   147   146   144   141
6  Российская федерация  Дальневосточный федеральный округ                Приморский край  2007  1953  1951  1947  1938  1933  1929  1923  1913  1902
7  Российская федерация  Дальневосточный федеральный округ             Республика Бурятия   967   972   971   972   974   978   982   984   985   983
8  Российская федерация  Дальневосточный федеральный округ        Республика Саха (Якутия)   954   958   956   956   955   957   960   963   964   967
9  Российская федерация  Дальневосточный федеральный округ            Сахалинская область   521   497   495   494   491   488   487   487   490   490
10 Российская федерация  Дальневосточный федеральный округ               Хабаровский край  1376  1343  1342  1342  1340  1338  1334  1333  1328  1321
```

Рассредоточение столбцов

```
df3_2 <- pivot_wider(
            data = df2,
            names_from = "Year",
            values_from = "Population"
)
```

```
> df3_2
# A tibble: 82 x 13
   Страна                Округ                          Область            X2005 X2010 X2011 X2012 X2013 X2014 X2015 X2016 X2017 X2018
   <chr>                 <chr>                          <chr>              <int> <int> <int> <int> <int> <int> <int> <int> <int> <int>
 1 Российская Федерация  Центральный федеральный округ  Белгородская область  1512  1532  1536  1541  1544  1548  1550  1553  1550  1548
 2 Российская Федерация  Центральный федеральный округ  Брянская область      1327  1275  1264  1254  1242  1233  1226  1221  1211  1200
 3 Российская Федерация  Центральный федеральный округ  Владимирская область  1486  1441  1432  1422  1413  1406  1397  1390  1378  1366
 4 Российская Федерация  Центральный федеральный округ  Воронежская область   2361  2335  2332  2330  2329  2331  2333  2335  2333  2328
 5 Российская Федерация  Центральный федеральный округ  Ивановская область    1102  1060  1054  1049  1043  1037  1030  1023  1015  1004
 6 Российская Федерация  Центральный федеральный округ  Калужская область     1023  1009  1008  1006  1005  1011  1010  1014  1012  1009
 7 Российская Федерация  Центральный федеральный округ  Костромская область    700   666   662   659   656   654   651   648   643   637
 8 Российская Федерация  Центральный федеральный округ  Курская область       1178  1126  1122  1119  1119  1117  1120  1123  1115  1107
 9 Российская Федерация  Центральный федеральный округ  Липецкая область      1194  1172  1166  1162  1160  1158  1156  1156  1150  1144
10 Российская Федерация  Центральный федеральный округ  Московская область    6784  7106  7199  7048  7134  7231  7319  7423  7503  7599
# ... with 72 more rows
```

Спецификации

Это новый функционал пакета tidyr, который ранее при работе с устаревшими функциями был недоступен.

Спецификация — это фрейм данных, каждая строка которого соответствует одному столбцу в новом выходном дата фрейме, и двумя специальными столбцами, которые начинаются с.:

- .name содержит исходное название столбца.
- .value содержит имя столбца, в который будут входить значения ячеек.

Остальные столбцы спецификации отражают то, как в новом столбце будет выводиться название сжимаемых столбцов из .name.

Спецификация описывает метаданные, хранящиеся в имени столбца, с одной строкой для каждого столбца и одним столбцом для каждой переменной, объединенной с именем столбца, наверное сейчас такое определение кажется запутанным, но после рассмотрения нескольких примеров всё станет значительно понятнее.

Смысл спецификации заключается в том, что вы можете извлекать, изменять и задавать новые метаданные к преобразуемому датафрейму.

Для работы со спецификациями при преобразовании таблицы из широкого формата в длинный служит функция pivot_longer_spec().

Спецификации

```
> df20 <-build_longer_spec(
            data = who,
            cols=new_sp_m014:newrel_f65,
            values_to = "count"
)
```

```
> df20
# A tibble: 56 x 3
   .name         .value name
   <chr>         <chr>  <chr>
 1 new_sp_m014   count  new_sp_m014
 2 new_sp_m1524  count  new_sp_m1524
 3 new_sp_m2534  count  new_sp_m2534
 4 new_sp_m3544  count  new_sp_m3544
 5 new_sp_m4554  count  new_sp_m4554
 6 new_sp_m5564  count  new_sp_m5564
 7 new_sp_m65    count  new_sp_m65
 8 new_sp_f014   count  new_sp_f014
 9 new_sp_f1524  count  new_sp_f1524
10 new_sp_f2534  count  new_sp_f2534
# ... with 46 more rows
```

```
df21 <- extract(
            data =df20,
            col = name,
            into = c("diagnosis", "gender", "age"),
            regex = "new_?(.*)_(.)(.*)"
)
```

```
> df21
# A tibble: 56 x 5
   .name         .value diagnosis gender age
   <chr>         <chr>  <chr>     <chr>  <chr>
 1 new_sp_m014   count  sp        m      014
 2 new_sp_m1524  count  sp        m      1524
 3 new_sp_m2534  count  sp        m      2534
 4 new_sp_m3544  count  sp        m      3544
 5 new_sp_m4554  count  sp        m      4554
 6 new_sp_m5564  count  sp        m      5564
 7 new_sp_m65    count  sp        m      65
 8 new_sp_f014   count  sp        f      014
 9 new_sp_f1524  count  sp        f      1524
10 new_sp_f2534  count  sp        f      2534
# ... with 46 more rows
```

Разделение столбцов

```
df4 <- separate(
        data = df1,
        col = "Округ",
        into = c("NewCol1","NewCol2","NewCol3","NewCol4"),
        sep = " "
)
```

По умолчанию разделитель – символ, не являющийся буквенно-цифровым

```
> df4
                Страна     NewCol1     NewCol2     NewCol3 NewCol4                  Область
1  Российская Федерация Центральный федеральный       округ    <NA>  Белгородская область
2  Российская Федерация Центральный федеральный       округ    <NA>      Брянская область
3  Российская Федерация Центральный федеральный       округ    <NA>  Владимирская область
4  Российская Федерация Центральный федеральный       округ    <NA>  Воронежская область
5  Российская Федерация Центральный федеральный       округ    <NA>   Ивановская область
6  Российская Федерация Центральный федеральный       округ    <NA>    Калужская область
7  Российская Федерация Центральный федеральный       округ    <NA>  Костромская область
8  Российская Федерация Центральный федеральный       округ    <NA>      Курская область
9  Российская Федерация Центральный федеральный       округ    <NA>     Липецкая область
10 Российская Федерация Центральный федеральный       округ    <NA>   Московская область
11 Российская Федерация Центральный федеральный       округ    <NA>     Орловская область
12 Российская Федерация Центральный федеральный       округ    <NA>     Рязанская область
13 Российская Федерация Центральный федеральный       округ    <NA>    Смоленская область
14 Российская Федерация Центральный федеральный       округ    <NA>    Тамбовская область
15 Российская Федерация Центральный федеральный       округ    <NA>      Тверская область
16 Российская Федерация Центральный федеральный       округ    <NA>      Тульская область
17 Российская Федерация Центральный федеральный       округ    <NA>   Ярославская область
18 Российская Федерация Центральный федеральный       округ    <NA>              г. Москва
19 Российская Федерация      Северо    Западный федеральный   округ  Республика Карелия
20 Российская Федерация      Северо    Западный федеральный   округ    Республика Коми
```

df4 <- separate(data = df1, col = "Округ", into = c("NewCol1",NA,"NewCol3","NewCol4"))

Объединение столбцов

```
df5 <- unite(
        data = df4,
        col = "New",
        "NewCol1", "NewCol2", "NewCol3", "NewCol4",
        sep = "*"
)
```

По умолчанию разделитель – нижнее подчеркивание

```
> df5
                Страна                                New                Область
1  Российская Федерация Центральный*федеральный*округ*NA Белгородская область
2  Российская Федерация Центральный*федеральный*округ*NA     Брянская область
3  Российская Федерация Центральный*федеральный*округ*NA  Владимирская область
4  Российская Федерация Центральный*федеральный*округ*NA   Воронежская область
5  Российская Федерация Центральный*федеральный*округ*NA   Ивановская область
6  Российская Федерация Центральный*федеральный*округ*NA    Калужская область
7  Российская Федерация Центральный*федеральный*округ*NA  Костромская область
8  Российская Федерация Центральный*федеральный*округ*NA      Курская область
9  Российская Федерация Центральный*федеральный*округ*NA     Липецкая область
10 Российская Федерация Центральный*федеральный*округ*NA    Московская область
```

Отсутствующие значения

	Страна	Округ	Область	X2005	X2010	X2011	X2012	X2013	X2014	X2015	X2016	X2017	X2018
30	Российская Федерация	Южный федеральный округ	Республика Калмыкия	294	289	287	284	282	281	279	278	275	272
31	Российская Федерация	Южный федеральный округ	Республика Крым	NA	NA	NA	NA	NA	1896	1907	1912	1914	1912
32	Российская Федерация	Южный федеральный округ	Краснодарский край	5127	5230	5284	5330	5404	5454	5514	5571	5603	5648
33	Российская Федерация	Южный федеральный округ	Астраханская область	1003	1010	1015	1014	1017	1021	1019	1019	1017	1014
34	Российская Федерация	Южный федеральный округ	Волгоградская область	2640	2607	2595	2583	2569	2557	2546	2535	2521	2508
35	Российская Федерация	Южный федеральный округ	Ростовская область	4332	4275	4260	4254	4246	4242	4236	4231	4221	4203
36	Российская Федерация	Южный федеральный округ	г. Севастополь	NA	NA	NA	NA	NA	399	416	429	437	443
37	Российская Федерация	Северо-Кавказский федеральный округ	Республика Дагестан	2693	2914	2931	2946	2964	2990	3015	3042	3064	3086

df2 <- gather(data = df1, "X2005","X2010","X2011","X2012","X2013","X2014","X2015","X2016","X2017","X2018", key = "Year", value = "Population")

30	Российская Федерация	Южный федеральный округ	Республика Калмыкия	X2005	294
31	Российская Федерация	Южный федеральный округ	Республика Крым	X2005	NA
32	Российская Федерация	Южный федеральный округ	Краснодарский край	X2005	5127
33	Российская Федерация	Южный федеральный округ	Астраханская область	X2005	1003
34	Российская Федерация	Южный федеральный округ	Волгоградская область	X2005	2640
35	Российская Федерация	Южный федеральный округ	Ростовская область	X2005	4332
36	Российская Федерация	Южный федеральный округ	г. Севастополь	X2005	NA
37	Российская Федерация	Северо-Кавказский федеральный округ	Республика Дагестан	X2005	2693

df6 <- gather(data = df1, "X2005","X2010","X2011","X2012","X2013","X2014","X2015","X2016","X2017","X2018", key = "Year", value = "Population", na.rm = TRUE)

29	Российская Федерация	Южный федеральный округ	Республика Адыгея	X2005	441
30	Российская Федерация	Южный федеральный округ	Республика Калмыкия	X2005	294
32	Российская Федерация	Южный федеральный округ	Краснодарский край	X2005	5127
33	Российская Федерация	Южный федеральный округ	Астраханская область	X2005	1003
34	Российская Федерация	Южный федеральный округ	Волгоградская область	X2005	2640
35	Российская Федерация	Южный федеральный округ	Ростовская область	X2005	4332

Отсутствующие значения

Функция complete() принимает набор столбцов и находит все уникальные комбинации. Далее она убеждается, что исходный набор данных содержит все эти значения, вставляя маркеры явного отсутствия значений (NA) там, где это необходимо.

df7 <- complete(data = df6,Страна,Округ,Область,Year)

```
64 Российская Федерация        Сибирский федеральный округ          Республика Хакасия   534   532   532   533   534   536   537
65 Российская Федерация        Сибирский федеральный округ              Алтайский край  2503  2417  2407  2399  2391  2385  2377
66 Российская Федерация        Сибирский федеральный округ            Красноярский край  2869  2829  2838  2847  2853  2859  2866
```

```
> df7
# A tibble: 6,560 x 5
   Страна               Округ                          Область        Year  Population
   <chr>                <chr>                          <chr>          <chr>      <int>
 1 Российская Федерация Дальневосточный федеральный округ Алтайский край X2005         NA
 2 Российская Федерация Дальневосточный федеральный округ Алтайский край X2010         NA
 3 Российская Федерация Дальневосточный федеральный округ Алтайский край X2011         NA
 4 Российская Федерация Дальневосточный федеральный округ Алтайский край X2012         NA
 5 Российская Федерация Дальневосточный федеральный округ Алтайский край X2013         NA
 6 Российская Федерация Дальневосточный федеральный округ Алтайский край X2014         NA
 7 Российская Федерация Дальневосточный федеральный округ Алтайский край X2015         NA
 8 Российская Федерация Дальневосточный федеральный округ Алтайский край X2016         NA
 9 Российская Федерация Дальневосточный федеральный округ Алтайский край X2017         NA
10 Российская Федерация Дальневосточный федеральный округ Алтайский край X2018         NA
# ... with 6,550 more rows
```

Отсутствующие значения

df8 <- complete(data = df6,Область,Year)

121	"г. Севастополь"	X2005	NA	NA	NA
122	"г. Севастополь"	X2010	NA	NA	NA
123	"г. Севастополь"	X2011	NA	NA	NA
124	"г. Севастополь"	X2012	NA	NA	NA
125	"г. Севастополь"	X2013	NA	NA	NA
126	"г. Севастополь"	X2014	Российская Федерация	Южный федеральный округ	399
127	"г. Севастополь"	X2015	Российская Федерация	Южный федеральный округ	416
128	"г. Севастополь"	X2016	Российская Федерация	Южный федеральный округ	429
129	"г. Севастополь"	X2017	Российская Федерация	Южный федеральный округ	437
130	"г. Севастополь"	X2018	Российская Федерация	Южный федеральный округ	443
531	"Республика Крым"	X2005	NA	NA	NA
532	"Республика Крым"	X2010	NA	NA	NA
533	"Республика Крым"	X2011	NA	NA	NA
534	"Республика Крым"	X2012	NA	NA	NA
535	"Республика Крым"	X2013	NA	NA	NA
536	"Республика Крым"	X2014	Российская Федерация	Южный федеральный округ	1896
537	"Республика Крым"	X2015	Российская Федерация	Южный федеральный округ	1907
538	"Республика Крым"	X2016	Российская Федерация	Южный федеральный округ	1912
539	"Республика Крым"	X2017	Российская Федерация	Южный федеральный округ	1914
540	"Республика Крым"	X2018	Российская Федерация	Южный федеральный округ	1912

Отсутствующие значения

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	**Страна**	**Округ**	**Область**	**2005**	**2010**	**2011**	**2012**	**2013**	**2014**	**2015**	**2016**	**2017**	**2018**
2	Российская Федерация	Центральный федеральный округ	Белгородская область	1512	1532	1536	1541	1544	1548	1550	1553	1550	1548
3			Брянская область	1327	1275	1264	1254	1242	1233	1226	1221	1211	1200
4			Владимирская область	1486	1441	1432	1422	1413	1406	1397	1390	1378	1366
5			Воронежская область	2361	2335	2332	2330	2329	2331	2333	2335	2333	2328
6			Ивановская область	1102	1060	1054	1049	1043	1037	1030	1023	1015	1004
7			Калужская область	1023	1009	1008	1006	1005	1011	1010	1014	1012	1009
8			Костромская область	700	666	662	659	656	654	651	648	643	637
9			Курская область	1178	1126	1122	1119	1119	1117	1120	1123	1115	1107
10			Липецкая область	1194	1172	1166	1162	1160	1158	1156	1156	1150	1144
11			Московская область	6784	7106	7199	7048	7134	7231	7319	7423	7503	7599
12			Орловская область	822	786	781	776	770	765	760	755	747	740
13			Рязанская область	1189	1152	1148	1144	1141	1135	1130	1127	1122	1114
14			Смоленская область	1025	983	981	975	968	965	959	953	950	942
15			Тамбовская область	1139	1090	1082	1076	1069	1062	1050	1040	1033	1016
16			Тверская область	1415	1350	1342	1334	1325	1315	1305	1297	1284	1270
17			Тульская область	1615	1550	1545	1532	1522	1514	1506	1499	1492	1479
18			Ярославская область	1313	1271	1271	1272	1272	1272	1272	1271	1266	1260
19			г. Москва	10924	11541	11613	11980	12108	12197	12330	12381	12507	12615
20		Северо-Западный федеральный округ	Республика Карелия	676	643	640	637	634	633	630	627	622	618
21			Республика Коми	963	899	890	880	872	864	857	850	841	830
22			Архангельская область	1282	1225	1213	1202	1192	1183	1174	1166	1155	1144
23			Вологодская область	1235	1201	1198	1196	1193	1191	1188	1184	1177	1168
24			Калининградская область	936	942	947	955	963	969	976	986	995	1002
25			Ленинградская область	1685	1719	1734	1751	1764	1776	1779	1792	1814	1848
26			Мурманская область	839	794	788	780	771	766	762	757	754	748
27			Новгородская область	666	633	630	626	623	619	616	613	606	600
28			Псковская область	721	671	667	662	657	651	646	642	636	630
29			г. Санкт-Петербург	4713	4899	4953	5028	5132	5192	5226	5282	5352	5384
30		Южный федеральный округ	Республика Адыгея	441	440	443	445	446	449	451	454	454	455
31			Республика Калмыкия	294	289	287	284	282	281	279	278	275	272
32			Республика Крым						1896	1907	1912	1914	1912
33			Краснодарский край	5127	5230	5284	5330	5404	5454	5514	5571	5603	5648
34			Астраханская область	1003	1010	1015	1014	1017	1021	1019	1019	1017	1014
35			Волгоградская область	2640	2607	2595	2583	2569	2557	2546	2535	2521	2508
36			Ростовская область	4332	4275	4260	4254	4246	4242	4236	4231	4221	4203
37			г. Севастополь						399	416	429	437	443

Отсутствующие значения

```
df10 <- read.csv2("C:\\Численность населения 2018 2.csv")
df10$Страна[df10$Страна==""] <- NA
df10$Округ[df10$Округ==""] <- NA
```

```
> df10
```

	Страна	Округ	Область	X2005	X2010	X2011	X2012	X2013	X2014	X2015
1	Российская Федерация	Центральный федеральный округ	Белгородская область	1512	1532	1536	1541	1544	1548	1550
2	<NA>	<NA>	Брянская область	1327	1275	1264	1254	1242	1233	1226
3	<NA>	<NA>	Владимирская область	1486	1441	1432	1422	1413	1406	1397
4	<NA>	<NA>	Воронежская область	2361	2335	2332	2330	2329	2331	2333
5	<NA>	<NA>	Ивановская область	1102	1060	1054	1049	1043	1037	1030
6	<NA>	<NA>	Калужская область	1023	1009	1008	1006	1005	1011	1010
7	<NA>	<NA>	Костромская область	700	666	662	659	656	654	651
8	<NA>	<NA>	Курская область	1178	1126	1122	1119	1119	1117	1120
9	<NA>	<NA>	Липецкая область	1194	1172	1166	1162	1160	1158	1156
10	<NA>	<NA>	Московская область	6784	7106	7199	7048	7134	7231	7319
11	<NA>	<NA>	Орловская область	822	786	781	776	770	765	760
12	<NA>	<NA>	Рязанская область	1189	1152	1148	1144	1141	1135	1130
13	<NA>	<NA>	Смоленская область	1025	983	981	975	968	965	959
14	<NA>	<NA>	Тамбовская область	1139	1090	1082	1076	1069	1062	1050
15	<NA>	<NA>	Тверская область	1415	1350	1342	1334	1325	1315	1305
16	<NA>	<NA>	Тульская область	1615	1550	1545	1532	1522	1514	1506
17	<NA>	<NA>	Ярославская область	1313	1271	1271	1272	1272	1272	1272
18	<NA>	<NA>	г. Москва	10924	11541	11613	11980	12108	12197	12330
19	<NA>	Северо-Западный федеральный округ	Республика Карелия	676	643	640	637	634	633	630
20	<NA>	<NA>	Республика Коми	963	899	890	880	872	864	857
21	<NA>	<NA>	Архангельская область	1282	1225	1213	1202	1192	1183	1174
22	<NA>	<NA>	Вологодская область	1235	1201	1198	1196	1193	1191	1188
23	<NA>	<NA>	Калининградская область	936	942	947	955	963	969	976
24	<NA>	<NA>	Ленинградская область	1685	1719	1734	1751	1764	1776	1779
25	<NA>	<NA>	Мурманская область	839	794	788	780	771	766	762
26	<NA>	<NA>	Новгородская область	666	633	630	626	623	619	616

Отсутствующие значения

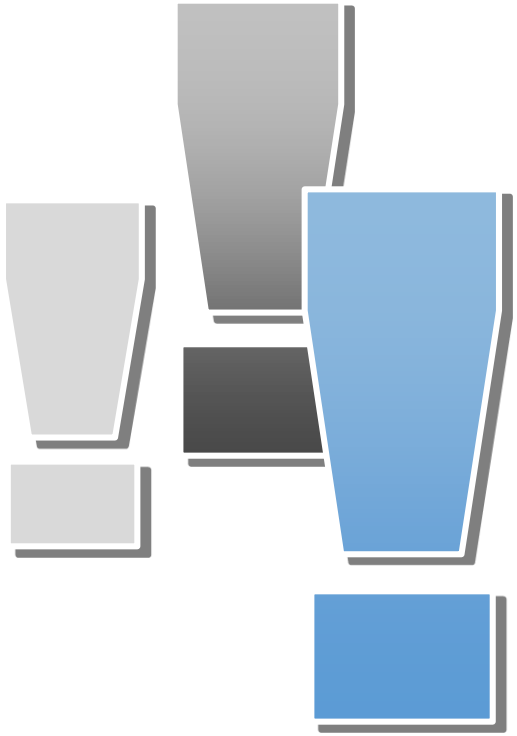
df11 <- fill(data = df10, Страна, Округ)

```
> df11
```

	Страна	Округ	Область	X2005	X2010	X2011	X2012	X2013	X2014	X2015
1	Российская Федерация	Центральный федеральный округ	Белгородская область	1512	1532	1536	1541	1544	1548	1550
2	Российская Федерация	Центральный федеральный округ	Брянская область	1327	1275	1264	1254	1242	1233	1226
3	Российская Федерация	Центральный федеральный округ	Владимирская область	1486	1441	1432	1422	1413	1406	1397
4	Российская Федерация	Центральный федеральный округ	Воронежская область	2361	2335	2332	2330	2329	2331	2333
5	Российская Федерация	Центральный федеральный округ	Ивановская область	1102	1060	1054	1049	1043	1037	1030
6	Российская Федерация	Центральный федеральный округ	Калужская область	1023	1009	1008	1006	1005	1011	1010
7	Российская Федерация	Центральный федеральный округ	Костромская область	700	666	662	659	656	654	651
8	Российская Федерация	Центральный федеральный округ	Курская область	1178	1126	1122	1119	1119	1117	1120
9	Российская Федерация	Центральный федеральный округ	Липецкая область	1194	1172	1166	1162	1160	1158	1156
10	Российская Федерация	Центральный федеральный округ	Московская область	6784	7106	7199	7048	7134	7231	7319
11	Российская Федерация	Центральный федеральный округ	Орловская область	822	786	781	776	770	765	760
12	Российская Федерация	Центральный федеральный округ	Рязанская область	1189	1152	1148	1144	1141	1135	1130
13	Российская Федерация	Центральный федеральный округ	Смоленская область	1025	983	981	975	968	965	959
14	Российская Федерация	Центральный федеральный округ	Тамбовская область	1139	1090	1082	1076	1069	1062	1050
15	Российская Федерация	Центральный федеральный округ	Тверская область	1415	1350	1342	1334	1325	1315	1305
16	Российская Федерация	Центральный федеральный округ	Тульская область	1615	1550	1545	1532	1522	1514	1506
17	Российская Федерация	Центральный федеральный округ	Ярославская область	1313	1271	1271	1272	1272	1272	1272
18	Российская Федерация	Центральный федеральный округ	г. Москва	10924	11541	11613	11980	12108	12197	12330
19	Российская Федерация	Северо-Западный федеральный округ	Республика Карелия	676	643	640	637	634	633	630
20	Российская Федерация	Северо-Западный федеральный округ	Республика Коми	963	899	890	880	872	864	857
21	Российская Федерация	Северо-Западный федеральный округ	Архангельская область	1282	1225	1213	1202	1192	1183	1174
22	Российская Федерация	Северо-Западный федеральный округ	Вологодская область	1235	1201	1198	1196	1193	1191	1188
23	Российская Федерация	Северо-Западный федеральный округ	Калининградская область	936	942	947	955	963	969	976
24	Российская Федерация	Северо-Западный федеральный округ	Ленинградская область	1685	1719	1734	1751	1764	1776	1779
25	Российская Федерация	Северо-Западный федеральный округ	Мурманская область	839	794	788	780	771	766	762
26	Российская Федерация	Северо-Западный федеральный округ	Новгородская область	666	633	630	626	623	619	616
27	Российская Федерация	Северо-Западный федеральный округ	Псковская область	721	671	667	662	657	651	646

.direction = c("down", "up", "downup", "updown")

Спасибо за внимание!



Шевцов Василий Викторович

shevtsov_vv@rudn.university
+7(903)144-53-57