JOINING DATA IN R WITH DPLYR

Welcome to the course!

Var_1	Var_2	Var_3	Var_4
obs_1	33	3	54
obs_2	20	90	22
obs_3	58	12	15
obs_4	83	81	5

```
> mean(df$Var_2)
[1] 48.5
```

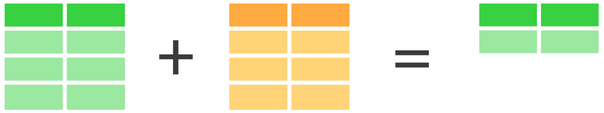
Var_1	Var_2	Var_3	Var_4	Var_5
obs_1	33	3	54	87
obs_2	20	90	22	42
obs_3	58	12	15	73
obs_4	83	81	5	88

```
> df$Var_5 <- df$Var_2 + df$Var_4
```

Course outline

- Chapter 1 - Mutating joins

- Chapter 2 - Filtering joins and set operations



- Chapter 3 - Assembling data

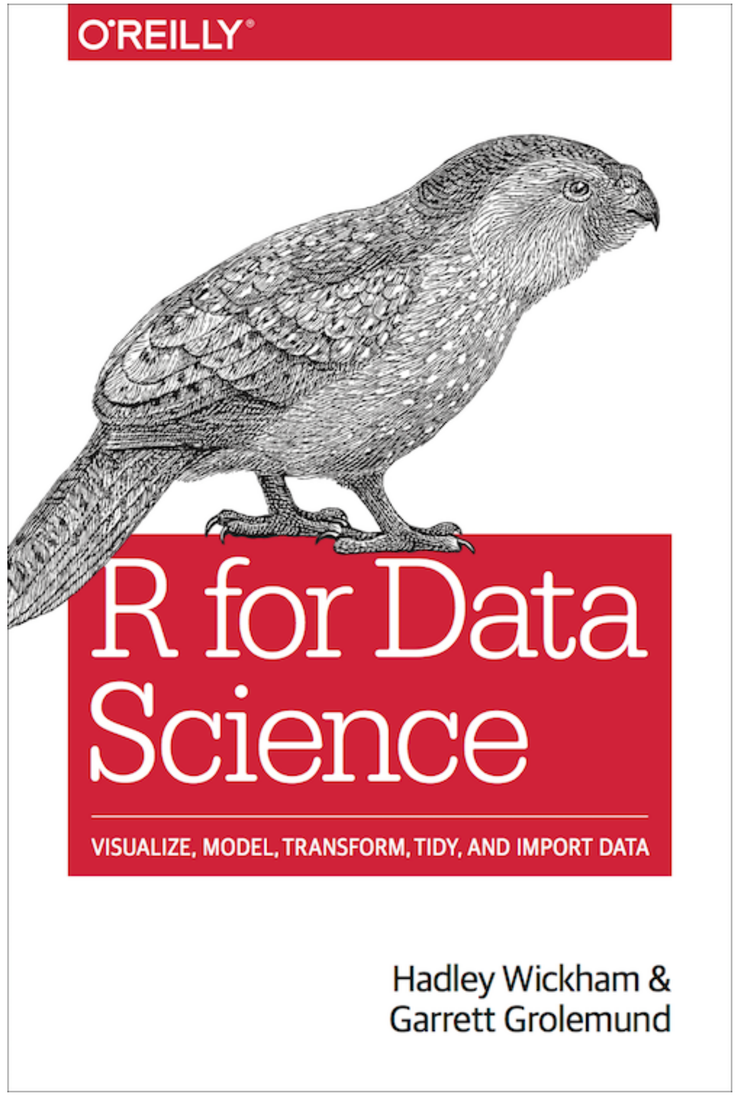
- Chapter 4 - Advanced joining
- Chapter 5 - Case study

- `arrange()`
- `filter()`
- `select()`
- `mutate()`
- `summarise()`

merge()

Benefits of dplyr join functions

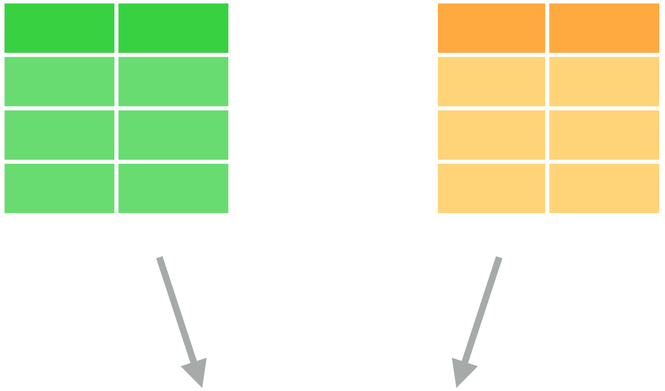
- Always preserve row order
- Intuitive syntax
- Can be applied to databases, spark, etc.


O'REILLY
R for Data Science
VISUALIZE, MODEL, TRANSFORM, TIDY, AND IMPORT DATA
Hadley Wickham &
Garrett Grolemund

Let’s practice!

JOINING DATA IN R WITH DPLYR

Keys



```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
# Example join output
   name    band  plays
1  Mick  Stones   <NA>
2  John Beatles Guitar
3  Paul Beatles   Bass
4 Keith    <NA> Guitar
```

Keys

primary key

foreign key

```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
# Example join output
   name    band  plays
1  Mick  Stones   <NA>
2  John Beatles Guitar
3  Paul Beatles   Bass
4 Keith    <NA> Guitar
```

Keys

primary key

foreign key

```
> names2
  name   surname    band
1 John  Coltrane      NA
2 John    Lennon Beatles
3 Paul McCartney Beatles
```

```
> plays2
   name   surname  plays
1  John    Lennon Guitar
2  Paul McCartney   Bass
3 Keith  Richards Guitar
```

```
# Example join output
   name   surname    band  plays
1  John  Coltrane    <NA>   <NA>
2  John    Lennon Beatles Guitar
3  Paul McCartney Beatles   Bass
4 Keith  Richards    <NA> Guitar
```

Let’s practice!

JOINING DATA IN R WITH DPLYR

Joins

left_join()

```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> left_join(names, plays, by = "name")

 name    band  plays
1 Mick  Stones   <NA>
2 John Beatles Guitar
3 Paul Beatles   Bass
```

rows from first table

values from second table

Multi-column keys

```
> names2
  name   surname    band
1 John  Coltrane      NA
2 John    Lennon Beatles
3 Paul McCartney Beatles
```

```
> plays2
   name   surname  plays
1  John    Lennon Guitar
2  Paul McCartney   Bass
3 Keith  Richards Guitar
```

```
> left_join(names2, plays2, by = c("name", "surname"))
```

```
  name   surname    band  plays
1 John  Coltrane    <NA>   <NA>
2 John    Lennon Beatles Guitar
3 Paul McCartney Beatles   Bass
```

right_join()

```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> right_join(names, plays, by = "name")
   name    band  plays
1  John Beatles Guitar
2  Paul Beatles   Bass
3 Keith    <NA> Guitar
```

rows from second table

values from first table

"tables"

- data frames
- tibbles (`tbl_df`)
- `tbl` references

tibble vs. data frame

```
> library(tibble)
> as.tibble(mtcars)
# A tibble: 32 × 11
     mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear
*  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1   21.0     6 160.0   110  3.90 2.620 16.46     0     1     4
2   21.0     6 160.0   110  3.90 2.875 17.02     0     1     4
3   22.8     4 108.0    93  3.85 2.320 18.61     1     1     4
4   21.4     6 258.0   110  3.08 3.215 19.44     1     0     3
5   18.7     8 360.0   175  3.15 3.440 17.02     0     0     3
6   18.1     6 225.0   105  2.76 3.460 20.22     1     0     3
7   14.3     8 360.0   245  3.21 3.570 15.84     0     0     3
8   24.4     4 146.7    62  3.69 3.190 20.00     1     0     4
9   22.8     4 140.8    95  3.92 3.150 22.90     1     0     4
10  19.2     6 167.6   123  3.92 3.440 18.30     1     0     4
# ... with 22 more rows, and 1 more variables: carb <dbl>
```

"tables"

- data frames
- tibbles (`tbl_df`)
- `tbl` references

Let’s practice!

JOINING DATA IN R WITH DPLYR

Mutating joins

mutate()

```
> pressure[1:4, ]
  temperature pressure
1           0   0.0002
2          20   0.0012
3          40   0.0060
4          60   0.0300
```

```
> mutate(pressure[1:4, ], fahrenheit = temperature * 1.8 + 32)
  temperature pressure fahrenheit
1           0   0.0002         32
2          20   0.0012         68
3          40   0.0060        104
4          60   0.0300        140
```

left_join()

```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> left_join(names, plays, by = "name")
  name    band  plays
1 Mick  Stones   <NA>
2 John Beatles Guitar
3 Paul Beatles   Bass
```

right_join()

```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> right_join(names, plays, by = "name")
   name    band  plays
1  John Beatles Guitar
2  Paul Beatles   Bass
3 Keith    <NA> Guitar
```

inner_join()

```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> inner_join(names, plays, by = "name")
  name    band  plays
1 John Beatles Guitar
2 Paul Beatles   Bass
```

full_join()

```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> full_join(names, plays, by = "name")
   name    band  plays
1  Mick  Stones   <NA>
2  John Beatles Guitar
3  Paul Beatles   Bass
4 Keith    <NA> Guitar
```

Syntax

```
> left_join( names, plays, by = "name")
> right_join(names, plays, by = "name")
> inner_join(names, plays, by = "name")
> full_join( names, plays, by = "name")
```

x y by

%>%

Pipe operator

```
> x <- 1:10

> x %>% sum()
[1] 55
```

```
> sum(x)
[1] 55
```

```
> abs(diff(range(x)))
[1] 9

> x %>%
>  range() %>%
>  diff() %>%
>  abs()
[1] 9
```

dplyr and pipes

```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> names %>%
+  full_join(plays, by = "name") %>%
+  mutate(missing_info = is.na(band) | is.na(plays)) %>%
+  filter(missing_info == TRUE) %>%
+  select(name, band, plays)
   name   band  plays
1  Mick Stones   <NA>
2 Keith   <NA> Guitar
```

Summary

- `left_join()`

- `right_join()`

- `inner_join()`

- `full_join()`

Let’s practice!