R Learners

6-27-2018

Topics

- Coding / re-coding
- Cleaning
- Reshaping
- Merging / joining

Re-coding

```
experiment %>%
  mutate(
    is_plausible = recode(Condition, 'Implausible' = FALSE, 'Plausible' = TRUE )
  )
```

Coding

```
experiment %>%
  ggplot(aes(RT)) +
  geom_histogram(bins = 100)
```

Coding

```
experiment %>%
  mutate(RT_group <- case_when(
      RT  > 1000 ~ 'High',
      RT <= 1000 ~ 'Normal')
  )
```

Cleaning

‘Work with Strings Cheat Sheet’

https://www.rstudio.com/resources/cheatsheets/

Cleaning

regular expressions (regex): “language for describing patterns in strings”

```
str_extract(string, pattern)
```

```
str_extract('xxx_123', '123')
```

```
str_extract('xxx_123', '[:digit:]')
```

```
str_extract('xxx_123', '[:digit:]+')
```

```
str_extract('xxx_123', '[:alpha:]')
```

```
str_extract('xxx_123', '[:alpha:]+')
```

Cleaning

```
df <- tibble(
  id = c('subject_1', 'subject_2', 'subject_3'),
  col_to_extract = c('xxx_123', 'xxx_456', 'xxx_789')
)
```

```
df %>% mutate(
  nums = str_extract(col_to_extract, '[:digit:]+'),
  chars = str_extract(col_to_extract, '[:alpha:]+')
)
```

Cleaning

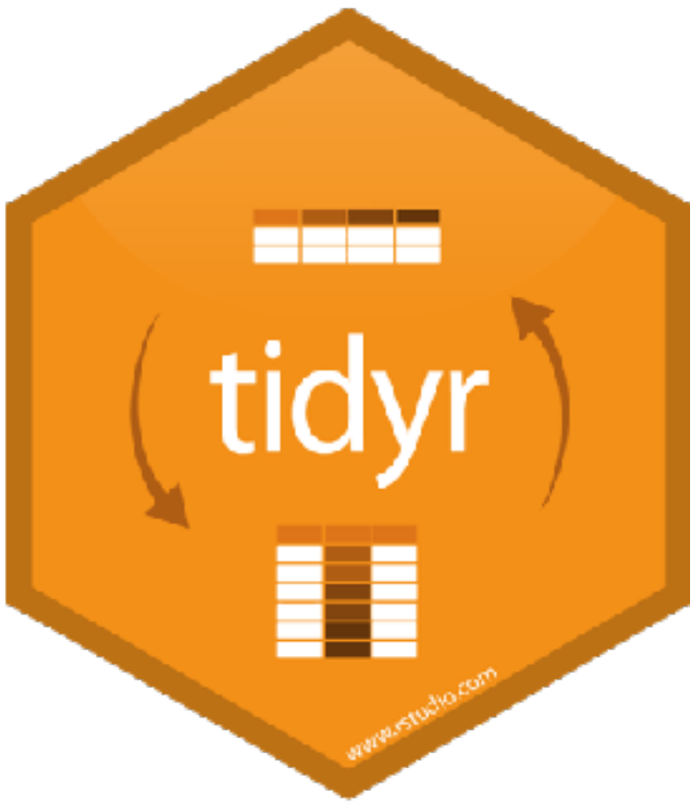
```
df %>%
  mutate(nums = str_extract(col_to_extract, '1[:digit:]+'))

df %>%
  mutate(nums = str_extract(col_to_extract, '[14][:digit:]+'))
```

Cleaning

```
df %>% mutate(
  nums = str_remove(col_to_extract, 'xxx_'),
  chars = str_remove(col_to_extract, '_[:digit:]+')
)
```

Reshaping



‘Data Import Cheat Sheet’

https://www.rstudio.com/resources/cheatsheets/

Reshaping

wide to long: `gather()`

“Gather takes multiple columns and collapses into key-value pairs.... Use `gather()` when you notice that you have columns that are not variables.”

country	1999	2000
A	0.7K	2K
B	37K	80K
C	212K	213K

→

country	year	cases
A	1999	0.7K
B	1999	37K
C	1999	212K
A	2000	2K
B	2000	80K
C	2000	213K

key value

Reshaping

```
df_gathered <- df %>%
  gather(`1999`, `2000`, key = "year", value = "cases")
```

country	1999	2000
A	0.7K	2K
B	37K	80K
C	212K	213K

country	year	cases
A	1999	0.7K
B	1999	37K
C	1999	212K
A	2000	2K
B	2000	80K
C	2000	213K

key value

Reshaping

long to wide: `spread()`

"`spread()` moves the unique values of a key column into the column names, spreading the values of a value column across the new columns "

country	year	cases
A	1999	0.7K
B	1999	37K
C	1999	212K
A	2000	2K
B	2000	80K
C	2000	213K

key value

country	1999	2000
A	0.7K	2K
B	37K	80K
C	212K	213K

Reshaping

```
df_gathered %>%
  spread(key = "year", value = "cases")
```

country	year	cases
A	1999	0.7K
B	1999	37K
C	1999	212K
A	2000	2K
B	2000	80K
C	2000	213K

key value

country	1999	2000
A	0.7K	2K
B	37K	80K
C	212K	213K

Merging / Joining

‘Data Transformation Cheat Sheet’

https://www.rstudio.com/resources/cheatsheets/

Merging / Joining

add variables: `bind_cols()`

* ensure rows correctly align

```
experiment_vars <- tibble(
  obs_id = 1:nrow(experiment)
)

experiment %>% bind_cols(experiment_vars)
```

Merging / Joining

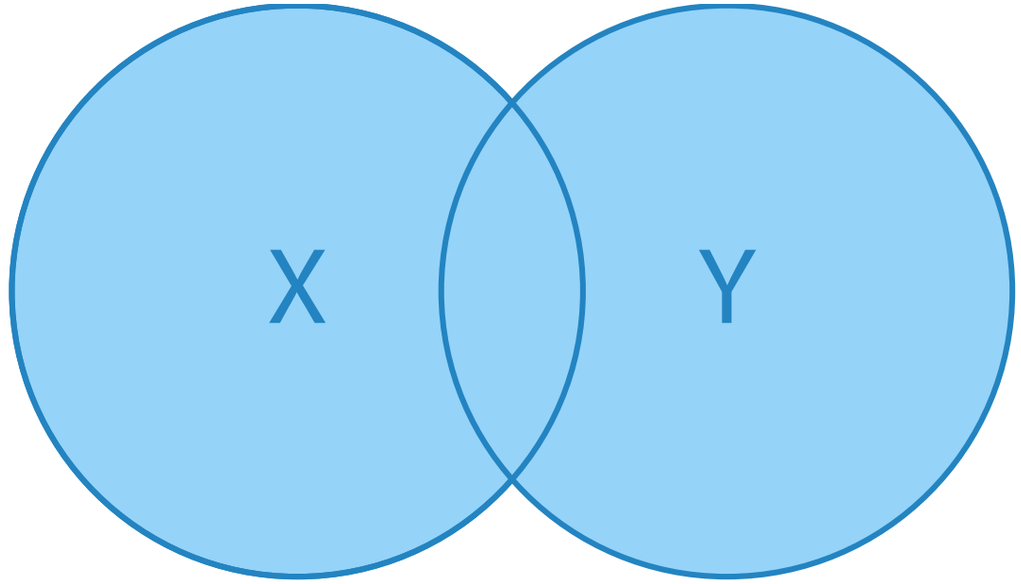
add observations: `bind_rows()`

```
experiment_obs <- tibble(
  Subject = 'S37',
  Age = '44',
  ItemName = 'Teacher',
  Condition = 'Plausible',
  TestingRoom = 3,
  SerialPosition = 1,
  RT = 654
)

experiment %>% bind_rows(experiment_obs)
```

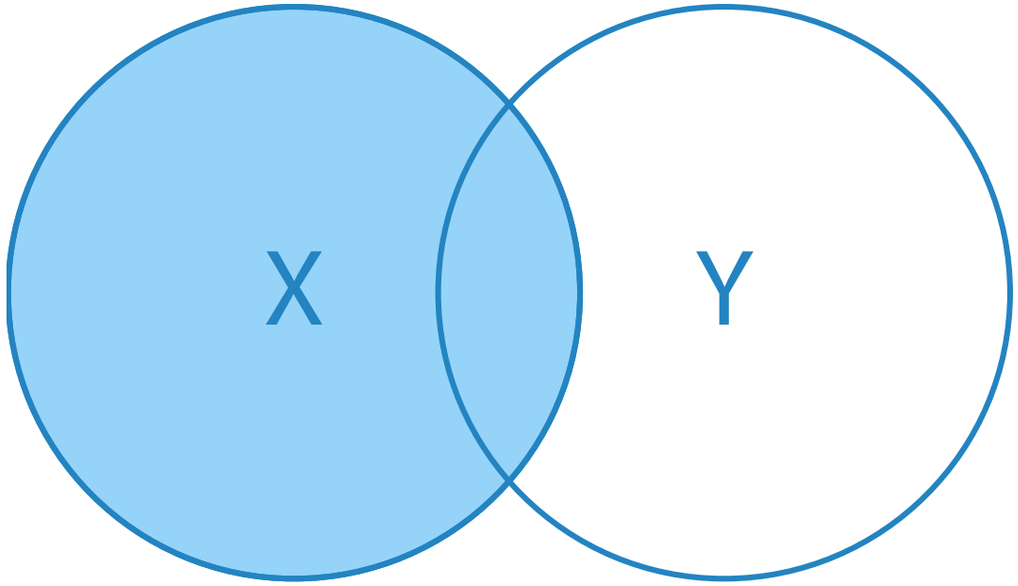
Merging / Joining

```
X %>% full_join(Y, by = 'id')
```



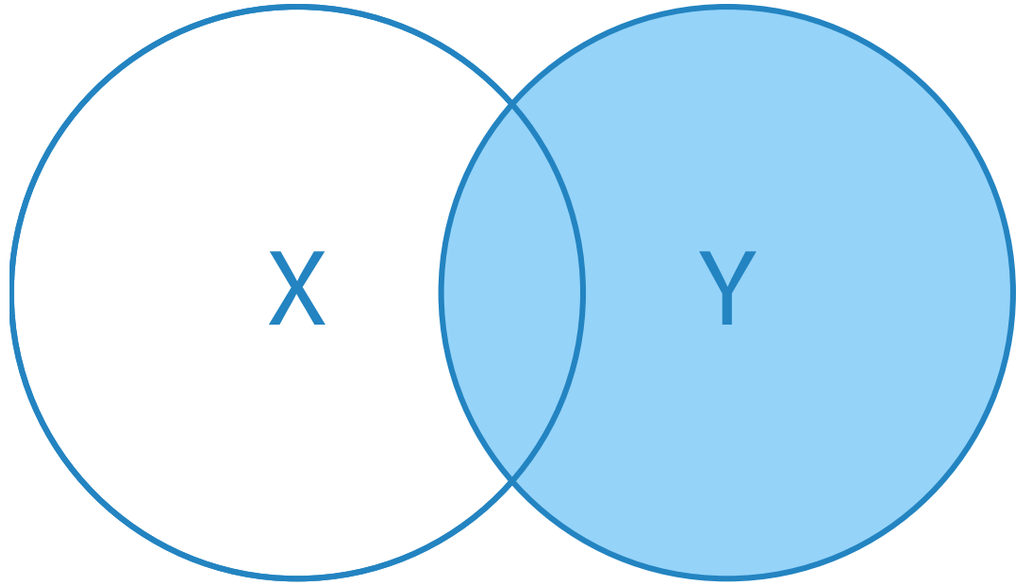
Merging / Joining

```
X %>% left_join(Y, by = 'id')
```



Merging / Joining

```
X %>% right_join(Y, by = 'id')
```



Merging / Joining

```
X %>% inner_join(Y, by = 'id')
```