Research Cycle 04: dplyr one-table verbs

Dale J. Barr

University of Glasgow

Tidy data

Wickham (2014)

"Happy families are all alike; every unhappy family is unhappy in its own way."

-Tolstoy

- Tidy datasets conform to a standardized way of linking data structure to data semantics (meaning)

Tidy data

(see also Codd, 1990; “3rd normal form”)

A dataset is a collection of values observed on variables across different observation units.

Definition (Tidy Data)

1. Each variable forms a column.
2. Each observation forms a row.
3. Each type of observational unit forms a table.

SubjectID	ItemID	Cond	RT	Choice
1	1	E	637	A
1	2	C	998	B
1	3	E	773	B
1	4	C	890	B
2	1	C	590	A
2	2	E	911	B
2	3	C	708	B
2	4	E	621	A

One of infinitely many messy versions

SubjectID	Cond1	Cond2	Cond3	Cond4	RT1	RT2	RT3	RT4	Ch1	Ch2	Ch3	Ch4
1	E	C	E	C	637	998	773	890	A	B	B	B
2	C	E	C	E	590	911	708	621	A	B	B	A

- wide format
- one column for each item for each variable, no easy mapping from structure to semantics
- column names different for same variable (e.g., RT1..RT4)
- different strategies for different obs units (e.g., calc subject means at each level of Cond)
  - must be done by hand, and thus, error prone

Tidy-ish representation of multilevel data

SubjectID	ListID	Gender	ItemID	Freq	TrialID	Cond	RT	Choice
1	X	F	1	L	1	E	637	A
1	X	F	2	H	2	C	998	B
1	X	F	3	L	3	E	773	B
1	X	F	4	H	4	C	890	B
2	Y	M	1	L	5	C	590	A
2	Y	M	2	H	6	E	911	B
2	Y	M	3	L	7	C	708	B
2	Y	M	4	H	8	E	621	A

- it obeys principles 1 & 2 (obs=rows, vars=cols), but violates 3
- PROBLEM: redundant information in the table, difficult to change values for certain variables, or add new variables at the subject level, error prone

Tidy representation of multilevel data

Subject

SubjectID	ListID	Gender
1	X	F
2	Y	M

Item

ItemID	Freq
1	H
2	L
3	H
4	L

Trial

SubjectID	ItemID	TrialID	Cond	RT	Choice
1	1	1	E	637	A
1	2	2	C	998	B
1	3	3	E	773	B
1	4	4	C	890	B
2	1	5	C	590	A
2	2	6	E	911	B
2	3	7	C	708	B
2	4	8	E	621	A

Tidy tools

Wickham (submitted)

Tidy tools

tidy data	$\rightarrow$	tidy data
input		*output*

transform create/modify variables, rearranging columns

filter include/exclude observations (rows)

aggregate collapse subsets of observations into single values

order sort observations

Not all tools in base `R` are tidy. Wickham's package `dplyr` adds tidy versions, plus additional functionality. Also, optimized for speed!

`dplyr` and the Wickham Six

According to R developer Hadley Wickham (@hadleywickham), 90% of data analysis can be reduced to the operations described by six English verbs.

`select()`	Include or exclude certain variables (columns)
`filter()`	Include or exclude certain observations (rows)
`mutate()`	Create new variables (columns)
`arrange()`	Change the order of observations (rows)
`group_by()`	Organize the observations into groups
`summarise()`	Derive aggregate variables for groups of observations

These functions reside in the add-on package `dplyr`. See the data wrangling cheat sheet!

Boolean expressions

Operator	Name	is TRUE if and only if
A < B	less than	A is less than B
A <= B	less than or equal	A is less than or equal to B
A > B	greater than	A is greater than B
A >= B	greater than or equal	A is greater than or equal to B
A == B	equivalence	A exactly equals B
A != B	not equal	A does not exactly equal B
A %in% B	in	A is an element of vector B