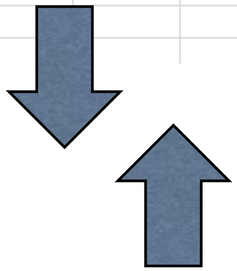
Transform Data with

row/column binding

rows: rbind() or dplyr::bind_rows()

StationCode	isSWMP	DateTimeStamp	Historical	Provisiona	CollMethd	REP	F_Record	PO4F	F_PO4F	NH4F	F_NH4F	N
gndbhnut	P	1/13/2016 12:05	0	1	1	1			<-2> [GDM] (CSM)		<-2> [GDM] (CSM)	
gndbhnut	P	1/13/2016 12:06	0	1	1	2			<-2> [GDM] (CSM)		<-2> [GDM] (CSM)	
gndbhnut	P	2/10/2016 8:51	0	1	1	1		0.002	<-4> [SBL]	0.038	<0>	
gndbhnut	P	11/16/2016 12:00	0	1	1	1		0.009	<0>	0.032	<0>	
gndbhnut	P	11/16/2016 12:01	0	1	1	2		0.014	<0>	0.038	<0>	
gndbhnut	P	12/14/2016 8:28	0	1	1	1		0.002	<0>	0.016	<1> [GSM] (CHB)	
gndbhnut	P	12/14/2016 8:29	0	1	1	2		0.002	<0>	0.016	<1> [GSM] (CHB)	

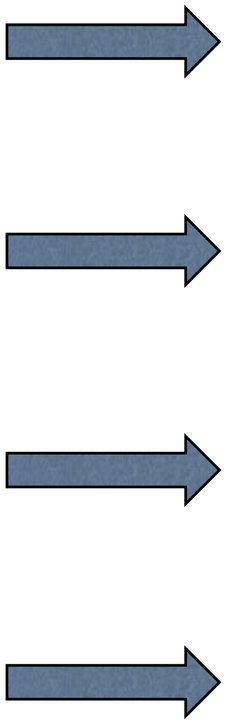


additional information as **additional rows**.

gndbhnut	P	1/17/2017 11:18	0	1	1	1		0.004	<0>	0.153	<0>
gndbhnut	P	1/17/2017 11:19	0	1	1	2		0.005	<0>	0.153	<0>
gndbhnut	P	2/13/2017 10:43	0	1	1	1		0.002	<-4> [SBL]	0.158	<0>
gndbhnut	P	2/13/2017 10:44	0	1	1	2		0.002	<-4> [SBL]	0.173	<0>
gndbhnut	P	3/13/2017 10:56	0	1	1	1		0.003	<0>	0.157	<0>
gndbhnut	P	3/13/2017 10:57	0	1	1	2		0.002	<0>	0.148	<0>
gndbhnut	P	4/10/2017 16:35	0	1	1	1		0.002	<0>	0.01	<0>
gndbhnut	P	4/10/2017 16:36	0	1	1	2		0.004	<0>	0.01	<0>
gndbhnut	P	5/8/2017 17:19	0	1	1	1		0.002	<-4> [SBL]	0.09	<0>
gndbhnut	P	5/8/2017 17:20	0	1	1	2		0.002	<0>	0.081	<0>
gndbhnut	P	6/19/2017 16:09	0	1	1	1		0.016	<0>	0.064	<0>
gndbhnut	P	6/19/2017 16:10	0	1	1	2		0.018	<0>	0.128	<0>
gndbhnut	P	7/19/2017 16:32	0	1	1	1		0.006	<0>	0.012	<0>

columns: cbind() or dplyr::bind_cols()

	year_sampled	salinity_ppt	water_temp_c
1	2005	8.1	23.3
2	2005	7.9	24.3
3	2005	8.2	24.8
4	2005	8.2	25.3
5	2005	9.0	24.8
6	2005	9.2	28.1
7	2005	8.4	26.2
8	2005	8.9	27.3
9	2005	11.5	27.6
10	2005	10.3	27.8
11	2005	10.4	26.5
12	2005	12.0	28.9
13	2005	10.2	27.6
14	2005	18.5	29.3



	Comp.1	Comp.2	Comp.3	Comp.4		
1	-1.469666048	0.50861098	-0.815343241	-1.46335107		
2	-1.755692184	0.24370146	-0.836290971	-2.18330031		
3	1.080570412	0.35374028	-2.236078623	-1.74776945		
4	-1.369822421	0.17215002	-1.135181748	-2.24956333		
5	-3.314759863	0.37444935	-0.137357002	-1.45224979		
6	-1.986256735	0.49716679	-1.157056717	-1.09732360		
7	-3.498663437	0.32469551	-0.265300898	-1.79427409		
8	-0.261967574	0.41546947	-1.878794797	-1.38602895		
9	-3.517448511	0.00847880	-0.318385366	-1.433961470	-0.480111879	0.517224945
10	-2.478236506	0.03892872	-0.867960703	-1.807396915	-0.268748165	0.130365940
11	1.766413104	0.23587234	-2.718299079	-1.189025121	-1.549863557	0.526441638
12	-1.401074695	0.04024652	-1.431812399	-1.186210314	-0.747486755	-0.207109698
13	-3.001089532	0.20084810	-0.598864512	-1.447290637	-0.701023350	0.275340991
14	-3.459846552	-0.54891989	-0.283326428	-0.239123537	-0.715070929	0.405253969

rows match
but
no identifying information for a join

- after PCA/nMDS
- after a ‘for’ loop

Joining Datasets

mutating joins

Motivating Example: Fish data

additional information
as **additional columns**
based on **matching information**

	A	B	C	D	E
1	site	habitat_type	lat	long	location
2	11	erosional edge	30.37163	-88.4438	Bayou Cumbest
3	14	erosional edge	30.3557	-88.4495	Pt aux Chens Bay
4	2	seagrass	30.38508	-88.4022	Middle Bay
5	3	seagrass	30.36205	-88.3977	Grand Bay
6	6	erosional edge	30.34905	-88.3973	Grand Battures
7	8	seagrass	30.35493	-88.4196	Jose Bay

	A	B	C	D	E	F	G
1	collection_id	site	season	year_sampled	salinity_ppt	do_mgl	water_temp_c
2	NFM08-142	2	Winter	2008	18.4	8.24	14
3	NFM08-143	3	Winter	2008	17.3	7.98	14.5
4	NFM08-146	6	Winter	2008	17.8	8.68	13.9
5	NFM08-148	8	Winter	2008	19.3	8.52	15.2
6	NFM08-151	11	Winter	2008	18.1	7.27	17.1
7	NFM08-154	14	Winter	2008	19.6	9.12	18.3
8	NFM08-156	2	Spring	2008	17.4	6.15	27.4
9	NFM08-157	3	Spring	2008	18.7	5.8	28.2
10	NFM08-160	6	Spring	2008	18.2	6.17	30
11	NFM08-162	8	Spring	2008	18.7	7.16	29.1
12	NFM08-165	11	Spring	2008	12.9	5.92	31.9
13	NFM08-168	14	Spring	2008	16.7	7.72	31.9
14	NFM08-169	2	Summer	2008	18.8	3.42	29.6
15	NFM08-170	3	Summer	2008	19.2	4.12	28.6
16	NFM08-173	6	Summer	2008	19.2	5.78	29.8
17	NFM08-175	8	Summer	2008	20.8	5.07	30.2
18	NFM08-178	11	Summer	2008	10.4	4.29	31.9
19	NFM08-180	14	Summer	2008	20.5	5.81	31.5

Motivating Example: Fish data

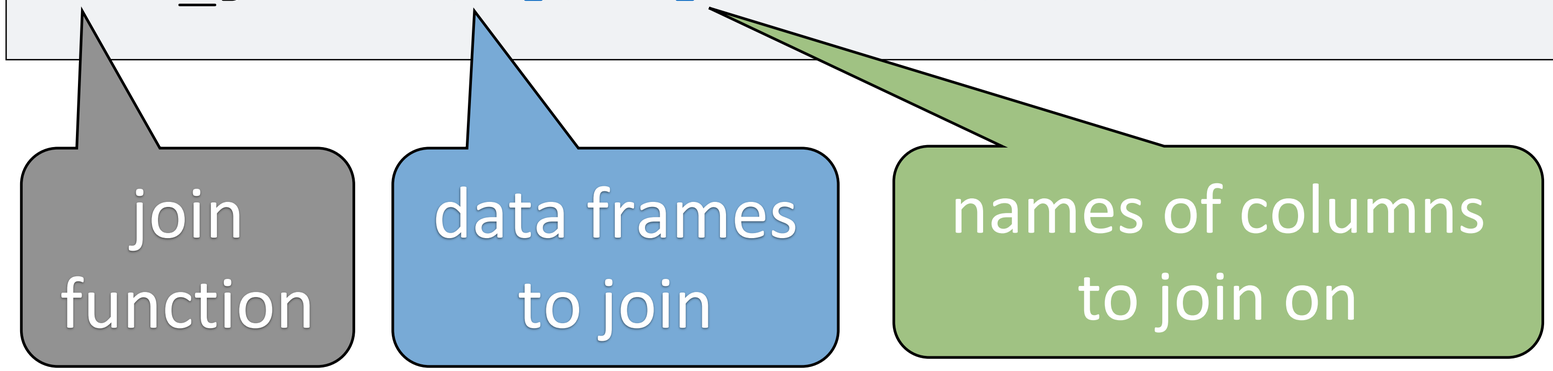
additional information
as **additional columns**
based on **matching information**

	collection_id	site	habitat_type	lat	long	season	year_sampled	salinity_ppt	do_mgl	water_temp_c
1	NFM08-142	2	seagrass	30.38508	-88.40215	Winter	2008	18.4	8.24	14.00
2	NFM08-143	3	seagrass	30.36205	-88.39772	Winter	2008	17.3	7.98	14.50
3	NFM08-146	6	erosional edge	30.34905	-88.39725	Winter	2008	17.8	8.68	13.90
4	NFM08-148	8	seagrass	30.35493	-88.41063	Winter	2008	19.3	8.52	15.20
5	NFM08-151	11	erosional edge	30.37163	-88.44382	Winter	2008	18.1	7.27	17.10
6	NFM08-154	14	erosional edge	30.35570	-88.44950	Winter	2008	19.6	9.12	18.30
7	NFM08-156	2	seagrass	30.38508	-88.40215	Spring	2008	17.4	6.15	27.40
8	NFM08-157	3	seagrass	30.36205	-88.39772	Spring	2008	18.7	5.80	28.20
9	NFM08-160	6	erosional edge	30.34905	-88.39725	Spring	2008	18.2	6.17	30.00

common syntax

Each join function returns a data frame / tibble.

```
left_join(x, y, by = NULL, … )
```



Toy data

```
band <- tribble(
   ~name,     ~band,
  "Mick",  "Stones",
  "John", "Beatles",
  "Paul", "Beatles"
)
```

band

name	band
Mick	Stones
John	Beatles
Paul	Beatles

```
instrument <- tribble(
    ~name,   ~plays,
   "John", "guitar",
   "Paul",   "bass",
  "Keith", "guitar"
)
```

instrument

name	plays
John	guitar
Paul	bass
Keith	guitar

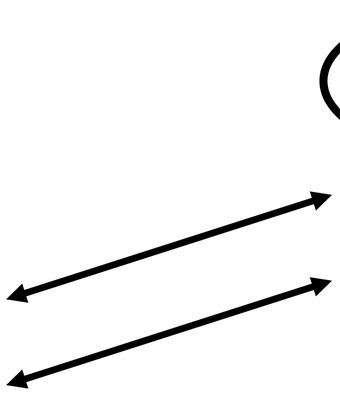
Toy data

band

name	band
Mick	Stones
John	Beatles
Paul	Beatles

instrument

name	plays
John	guitar
Paul	bass
Keith	guitar



left

```
band %>% left_join(instrument, by = "name")
```

band

name	band
Mick	Stones
John	Beatles
Paul	Beatles

+

instrument

name	plays
John	guitar
Paul	bass
Keith	guitar

=

name	band	plays
Mick	Stones	<NA>
John	Beatles	guitar
Paul	Beatles	bass

right

```
band %>% right_join(instrument, by = "name")
```

band

name	band
Mick	Stones
John	Beatles
Paul	Beatles

+

instrument

name	plays
John	guitar
Paul	bass
Keith	guitar

=

name	band	plays
John	Beatles	guitar
Paul	Beatles	bass
Keith	<NA>	guitar

full

```
band %>% full_join(instrument, by = "name")
```

band

name	band
Mick	Stones
John	Beatles
Paul	Beatles

+

instrument

name	plays
John	guitar
Paul	bass
Keith	guitar

=

name	band	plays
Mick	Stones	<NA>
John	Beatles	guitar
Paul	Beatles	bass
Keith	<NA>	guitar

inner

```
band %>% inner_join(instrument, by = "name")
```

band

name	band
Mick	Stones
John	Beatles
Paul	Beatles

+

instrument

name	plays
John	guitar
Paul	bass
Keith	guitar

=

name	band	plays
John	Beatles	guitar
Paul	Beatles	bass

What if the names do not match?

Use a named vector to match on variables with different names.

```
df1 %>% left_join(df2, by = c("df1_col" = "df2_col"))
```

A named vector

The name of the element = the column name in the first data set

The value of the element = the column name in the second data set

Toy data

```
band <- tribble(
   ~name,     ~band,
  "Mick",  "Stones",
  "John", "Beatles",
  "Paul", "Beatles"
)
```

band

name	band
Mick	Stones
John	Beatles
Paul	Beatles

```
instrument2 <- tribble(
    ~artist,   ~plays,
   "John", "guitar",
   "Paul",   "bass",
  "Keith", "guitar"
)
```

instrument2

artist	plays
John	guitar
Paul	bass
Keith	guitar

nonmatching names

```
band %>% left_join(instrument2, by = c("name" = "artist"))
```

band

name	band
Mick	Stones
John	Beatles
Paul	Beatles

+

instrument2

artist	plays
John	guitar
Paul	bass
Keith	guitar

=

name	band	plays
Mick	Stones	<NA>
John	Beatles	guitar
Paul	Beatles	bass

Recap: Two table verbs

left_join() retains all cases in **left** data set

right_join() retains all cases in **right** data set

full_join() retains all cases in **either** data set

inner_join() retains only cases in **both** data sets

Two table verbs