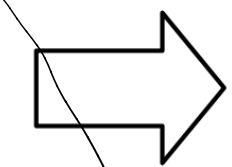
Reshaping data

Logistics and reminders

- HW 1 due tonight
- HW 2 released, due next Friday
- Department seminar coming up on 9/11
  - 11am in ZSR auditorium
  - Speaker: Robert Langefeld
  - Attendance part of class participation grade
  - If you can’t attend in person, can instead watch a seminar on YouTube

Last time: `pivot_longer`

id	bp1	bp2
A	100	120
B	140	115
C	120	125



id	measurement	value
A	bp1	100
A	bp2	120
B	bp1	140
B	bp2	115
C	bp1	120
C	bp2	125

```
df |>
  pivot_longer(
    cols = bp1:bp2,
    names_to = "measurement",
    values_to = "value"
  )
```

Why pivot?

```
# A tibble: 260 × 38
   country `1975` `1976` `1977` `1978` `1979` `1980` `1981` `1982`
`1983` `1984`
   <chr>    <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
<dbl>  <dbl>
 1 Afghan…     NA     NA     NA     NA   4.99   NA        NA     NA
NA   NA
 2 Albania     NA     NA     NA     NA  NA      NA        NA     NA
NA   NA
 3 Algeria     NA     NA     NA     NA  NA      NA        NA     NA
NA   NA
 4 Andorra     NA     NA     NA     NA  NA      NA        NA     NA
NA   NA
```

Challenge: a variable of interest (year) is contained in the column names!

Why pivot?

Literacy data in narrow form:

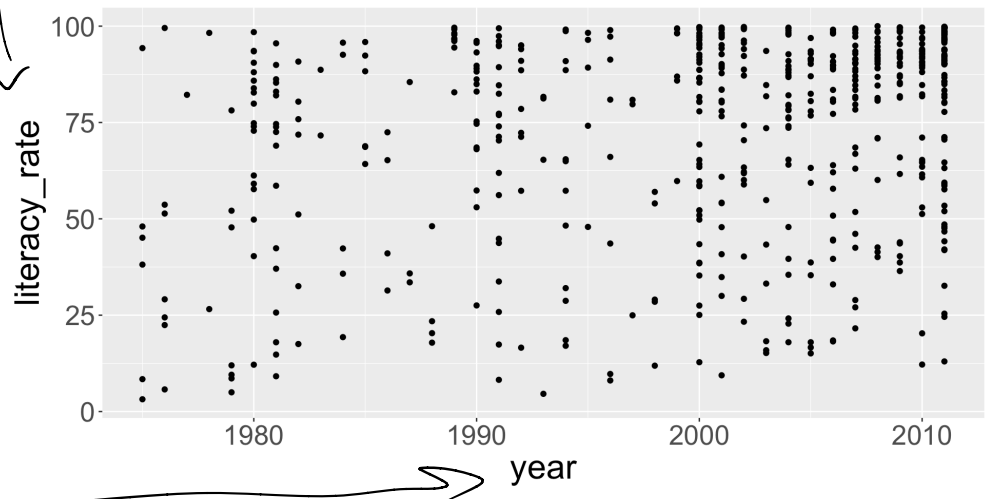
```r
litF_long <- litF |>
  pivot_longer(
    cols = -country,
    names_to = "year",
    values_to = "literacy_rate",
    values_drop_na = T
  )

litF_long
```

```
# A tibble: 571 × 3
   country     year  literacy_rate
   <chr>       <chr>         <dbl>
 1 Afghanistan 1979           4.99
 2 Afghanistan 2011          13
 3 Albania     2001          98.3
 4 Albania     2008          94.7
 5 Albania     2011          95.7
 6 Algeria     1987          35.8
```

Why pivot?

	country	year	literacy_rate
1	Afghanistan	1979	4.987460
2	Afghanistan	2011	13.000000
3	Albania	2001	98.252274
4	Albania	2008	94.681814
5	Albania	2011	95.691480
6	Algeria	1987	35.839915
7	Algeria	2002	60.075082
8	Algeria	2006	63.918785
9	Angola	2001	54.194488
10	Angola	2011	58.608460
11	Anguilla	1984	95.714930
12	Antigua and Barbuda	2001	99.420000
13	Antigua and Barbuda	2011	99.420000
14	Argentina	1980	93.580894
15	Argentina	1991	96.041358
16	Argentina	2001	97.193411




```
litF_long |>
  ggplot(aes(x = year,
             y = literacy_rate)) +
  geom_point()
```

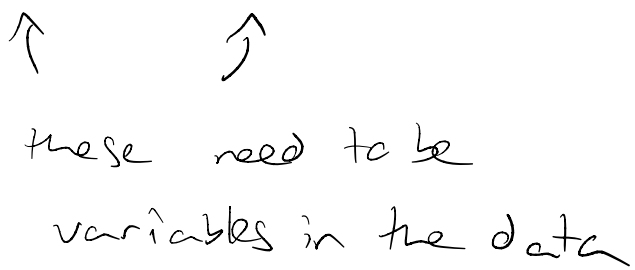
each piece of the plot needs to be a variable in the data

Why pivot?

	country	year	literacy_rate
1	Afghanistan	1979	4.987460
2	Afghanistan	2011	13.000000
3	Albania	2001	98.252274
4	Albania	2008	94.681814
5	Albania	2011	95.691480
6	Algeria	1987	35.839915
7	Algeria	2002	60.075082
8	Algeria	2006	63.918785
9	Angola	2001	54.194488
10	Angola	2011	58.608460
11	Anguilla	1984	95.714930
12	Antigua and Barbuda	2001	99.420000
13	Antigua and Barbuda	2011	99.420000
14	Argentina	1980	93.580894
15	Argentina	1991	96.041358
16	Argentina	2001	97.193411

(Intercept)	year
-1323.2098674	0.6979597

```
1 lm(literacy_rate ~ year, data = litF_long)
```



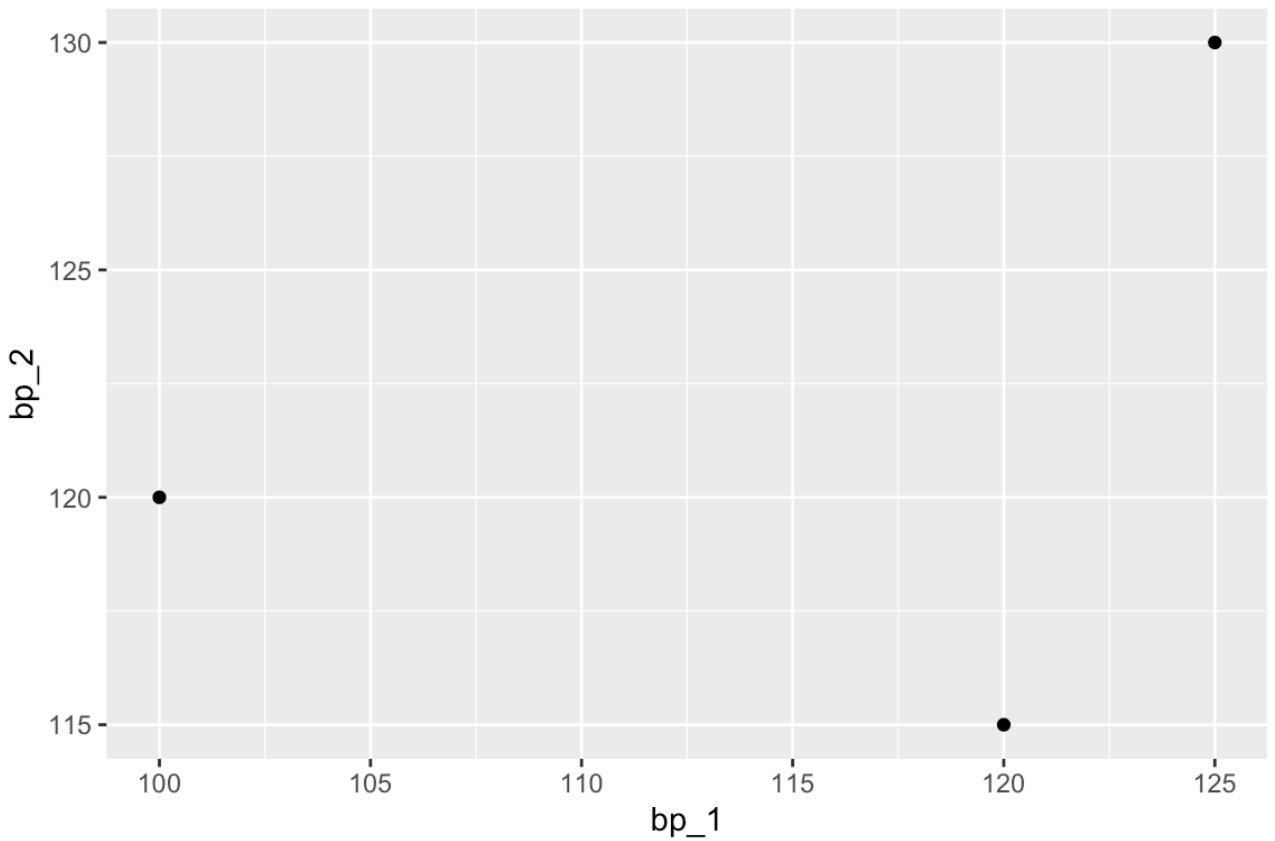
Another example from last time

```
1 df_3
```

```
  id bp_1 bp_2 hr_1 hr_2
1  1  100  120   60   77
2  2  120  115   75   81
3  3  125  130   80   93
```

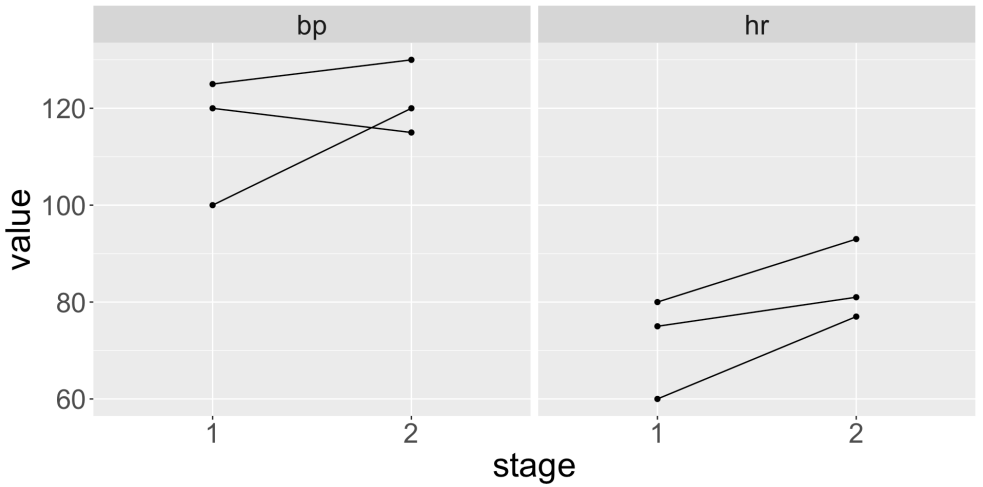
What we can do with the current data

```
df_3 |>
  ggplot(aes(x = bp_1,
             y = bp_2)) +
  geom_point()
```



What we can do with reshaped data

```
# A tibble: 12 × 4
      id measurement stage value
   <dbl> <chr>       <chr> <dbl>
 1     1 bp          1       100
 2     1 bp          2       120
 3     1 hr          1        60
 4     1 hr          2        77
 5     2 bp          1       120
 6     2 bp          2       115
 7     2 hr          1        75
 8     2 hr          2        81
 9     3 bp          1       125
10     3 bp          2       130
```




```
df3_long |>
  ggplot(aes(x = stage,
             y = value)) +
  geom_point() +
  geom_line(aes(group = id)) +
  facet_wrap(~measurement)
```

Warmup activity

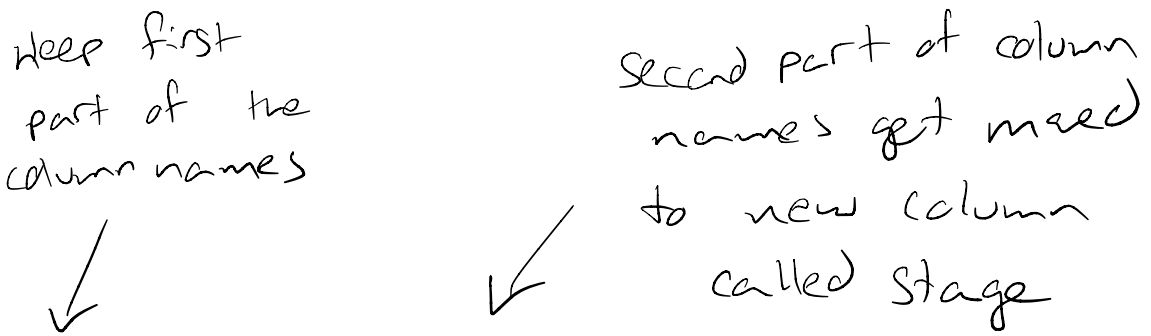
Work on the activity (handout) with a neighbor, then we will discuss as a class

Warmup

```
1 df_3
2
3 df_3 |>
4   pivot_longer(cols = -id,
5                names_to = c(".value", "stage"),
6                names_sep = "_")
```

```
  id bp_1 bp_2 hr_1 hr_2
1  1  100  120   60   77
2  2  120  115   75   81
3  3  125  130   80   93
```

```
  id stage bp hr
1  1     1 100 60
2  1     2 120 77
3  2     1 120 75
4  2     2 115 81
5  3     1 125 80
6  3     2 130 93
```



What is `names_to = c(".value", "stage")` doing?

What we can do with the reshaped data

```
# A tibble: 6 × 4
     id stage    bp    hr
  <dbl> <chr> <dbl> <dbl>
1     1 1       100    60
2     1 2       120    77
3     2 1       120    75
4     2 2       115    81
5     3 1       125    80
6     3 2       130    93
```

```
Call:
lm(formula = bp ~ hr + stage, data =
df3_new)

Coefficients:
(Intercept)           hr       stage2
     38.022        1.074       -6.223
```

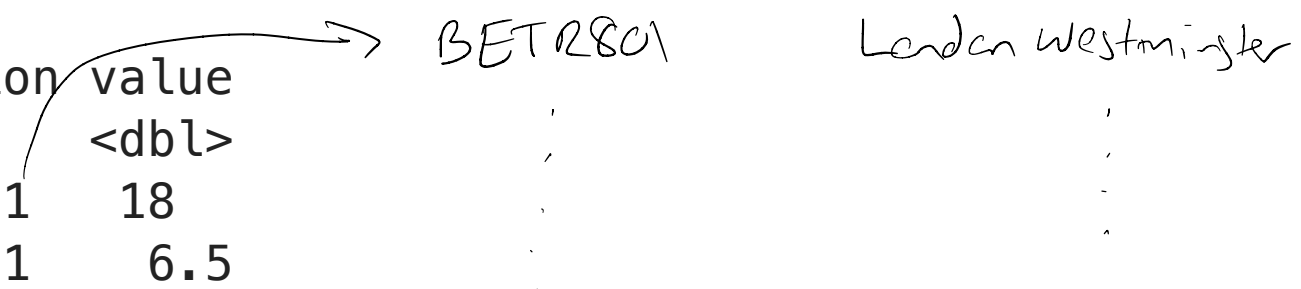
```r
lm(bp ~ hr + stage,
   data = df3_new)
```

Going the other way

Data on air quality in two locations (BETR801, London Westminster) on different days:

```
1 air_quality
```

```
# A tibble: 1,825 × 3
   date.utc            location value
   <dttm>              <chr>    <dbl>
 1 2019-06-18 06:00:00 BETR801   18
 2 2019-06-17 08:00:00 BETR801    6.5
 3 2019-06-17 07:00:00 BETR801   18.5
 4 2019-06-17 06:00:00 BETR801   16
 5 2019-06-17 05:00:00 BETR801    7.5
 6 2019-06-17 04:00:00 BETR801    7.5
 7 2019-06-17 03:00:00 BETR801    7
 8 2019-06-17 02:00:00 BETR801    7
 9 2019-06-17 01:00:00 BETR801    8
10 2019-06-16 01:00:00 BETR801   15
```



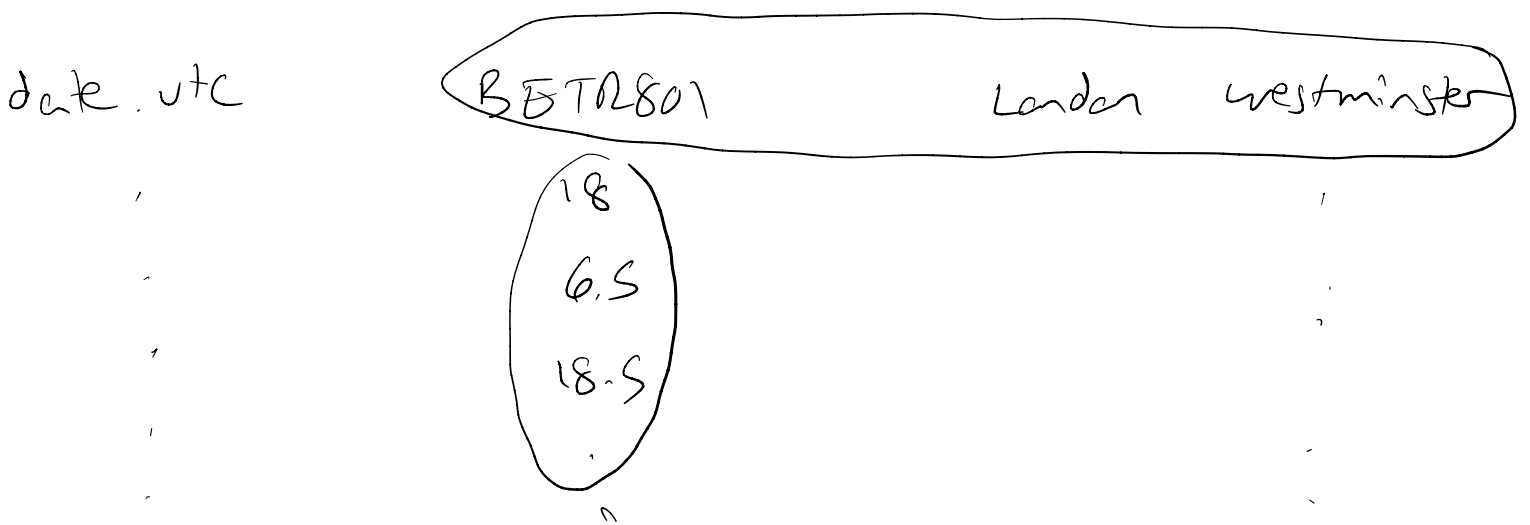
What if I want a separate column for each location?

pivot_wider

```
# A tibble: 3 × 3
  date.utc             location value
  <dttm>               <chr>    <dbl>
1 2019-06-18 06:00:00 BETR801   18
2 2019-06-17 08:00:00 BETR801    6.5
3 2019-06-17 07:00:00 BETR801   18.5
```

```
1 air_quality |>
2   pivot_wider(id_cols = ...,
3               names_from = ...,
4               values_from = ...)
```

<- columns to keep (date.utc)
<- new column names (location)
<- values to fill in (value)



pivot_wider

```
# A tibble: 3 × 3
  date.utc            location value
  <dttm>              <chr>    <dbl>
1 2019-06-18 06:00:00 BETR801   18
2 2019-06-17 08:00:00 BETR801    6.5
3 2019-06-17 07:00:00 BETR801   18.5
```

```
air_quality |>
  pivot_wider(id_cols = date.utc,
              names_from = location,
              values_from = value)
```

```
# A tibble: 1,670 × 3
   date.utc            BETR801 `London Westminster`
   <dttm>                <dbl>                <dbl>
 1 2019-06-18 06:00:00    18                      7
 2 2019-06-17 08:00:00     6.5                    6
 3 2019-06-17 07:00:00    18.5                    6
 4 2019-06-17 06:00:00    16                      6
 5 2019-06-17 05:00:00     7.5                    6
 6 2019-06-17 04:00:00     7.5                    6
 7 2019-06-17 03:00:00     7                      6
```

Class activity

https://sta279-f25.github.io/class_activities/ca_05.html

- Work with a neighbor on the class activity
- At the end of class, submit your work as an HTML file on Canvas (one per group, list all your names)

For next time, read:

- Chapter 5 in *Modern Data Science with R*