

Factors and Pivoting in R

Bio724D: Fall 2023

2023-10-01

Data for examples

Yeast colony morphology related data from Granek et al. 2013

```
data_URL <- "https://tinyurl.com/36h67mhm"  
yeast <- read_csv(data_URL)
```

Yeast data

	Strain	Segregant	Pool	Cyr1.geno	Flo11.geno	CM.a	CM.b	CM.c	cAMP	Cyr1.expr	Flo11.expr	Adhes.a	Adhes.b	Adhes.c
1	PMY1278	s44	C	C	S	4.3	3.9	3.9	221.34	1816	226922	0.4055	0.4576	0.4564
2	PMY1330	s67	C	C	C	3.6	3.6	3.5	276.70	NA	NA	0.4989	0.9868	1.0785
3	PMY1331	s68	C	C	C	3.2	3.2	3.5	122.06	NA	NA	0.5176	1.2117	1.0973
4	PMY1254	s20	S	S	S	1.0	1.0	1.0	164.80	1418	22337	0.0576	0.3082	0.2970
5	PMY1289	s55	C	S	C	3.1	3.0	3.6	195.59	2159	184194	0.3301	0.4025	0.4269
6	PMY1255	s21	S	S	S	1.0	1.0	1.0	252.33	1555	156586	0.1703	0.1821	0.1847
7	PMY1294	s60	C	C	C	3.3	3.6	3.7	247.58	1864	136993	0.7280	1.5931	1.4825
8	PMY1288	s54	C	C	C	3.6	3.7	3.5	197.00	1140	136704	0.4053	0.6482	0.6894
9	PMY1260	s26	S	S	S	1.0	1.0	1.0	194.97	1444	2113	0.0887	0.1066	0.1055
10	PMY1264	s30	S	S	S	1.0	1.0	1.0	176.29	1265	82581	0.1414	0.2100	0.1957

Figure 1: A sample of rows from seg_strain_table.csv

Factors

- Factors represent are used categorical variables in R. They are a unique data type that under hood maps character strings to integers.
- Factors can be ordered or unordered

Creating factors from scratch

```
# unordered factor
flavors <- factor(c("sweet", "sour", "sweet", "salty"))

# ordered factor
sizes <- factor(c("small", "small", "tiny", "large", "medium"),
               levels = c("tiny", "small", "medium", "large"),
               ordered = TRUE)
```

flavors

```
[1] sweet sour  sweet salty
Levels: salty sour sweet
```

sizes

```
[1] small  small  tiny   large  medium
Levels: tiny < small < medium < large
```

Factors: Your turn

What behaviors do you observe in the examples below?

```
flavors <- factor(c("sweet","sour", "sweet", "salty"))  
  
sizes <- factor(c("small", "small", "tiny", "large", "medium"),  
               levels = c("tiny", "small", "medium", "large"),  
               ordered = TRUE)
```

- Example A

```
flavors <- factor(c("sweet","sour","salty", "sour"))  
flavors[3] <- "umami"
```

- Example B

```
sizes < "medium"
```

- Example C

```
flavors < "sweet"
```

Factors from continuous data

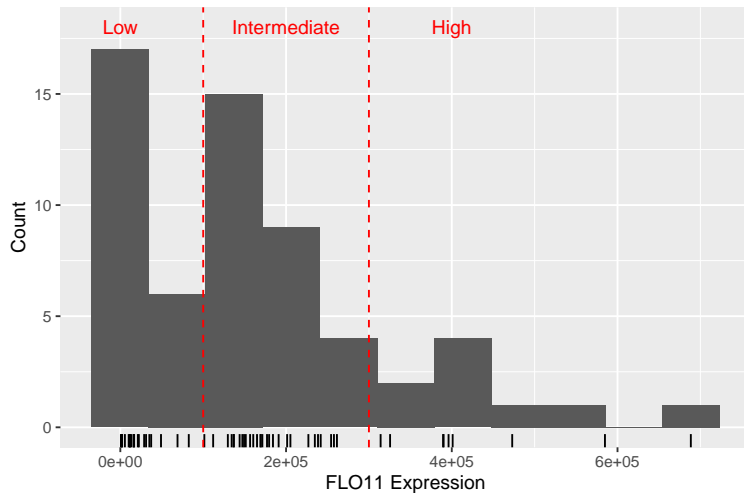
```
x <- 1:10  
x
```

```
[1] 1 2 3 4 5 6 7 8 9 10
```

```
factored_x <- cut(x,  
                 breaks = c(0, 4, 7, 10),  
                 labels = c("small", "medium", "large"),  
                 ordered_result = TRUE)  
factored_x
```

```
[1] small small small small medium medium medium large large large  
Levels: small < medium < large
```

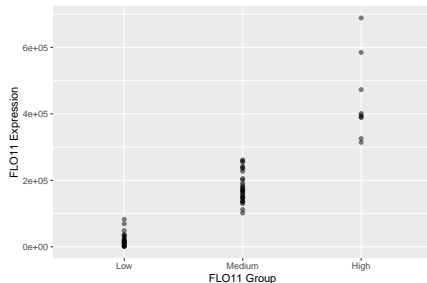
Factoring example



Factoring example: Your turn

Complete the following code to create an ordered factor, and corresponding figure, representing FLO11 expression categories:

```
yeast |>
  mutate(Flo11.group = cut(
    ## your code here
  ) |>
  filter(!is.na(Flo11.group)) |>
  ggplot(aes(x = Flo11.group, y = Flo11.expr)) +
  geom_point(alpha=0.5) +
  labs(x = "FLO11 Factor", y = "FLO11 Expression")
```



Pivoting

Pivoting is the act of reshaping our data to make it “longer” or “wider”

- longer = fewer columns, more rows
 - Pivoting to make our data longer is frequently done when there is information in column names that we want to treat as variable values for the purposes of grouping, plotting, faceting, etc.
- wider = more columns, fewer rows
 - Pivoting to make our data wider is usually done when it's useful to take entries in one or more columns and turn those into column headers (variable names) and take corresponding entries from other columns and make those the values in the new column

Pivot longer example

- `tidyr::pivot_longer` is the core function for long pivoting

```
cm_example <-  
  yeast |>  
  select(Strain, Pool, CM.a:CM.c) |>  
  slice_sample(n=2)
```

Strain	Pool	CM.a	CM.b	CM.c
PMY1277	C	2.8	2.8	3.3
PMY1278	C	4.3	3.9	3.9

```
long_cm <-  
  cm_example |>  
  pivot_longer(starts_with("CM."),  
               names_to = c("Replicate"),  
               values_to = c("Value"))
```

Strain	Pool	Replicate	Value
PMY1277	C	CM.a	2.8
PMY1277	C	CM.b	2.8
PMY1277	C	CM.c	3.3
PMY1278	C	CM.a	4.3
PMY1278	C	CM.b	3.9
PMY1278	C	CM.c	3.9

Pivot wider example