

Lab 4: Tidying your dataset

Karena Andrea Moreno-Acero

2024-11-01

Prune the dataset

Exercise 1

Enforcing one observation per row

```
brauer2 <- brauer %>% select(-GID, -YORF, -GWEIGHT)
```

Exercise 2

```
brauer3 <- brauer2 %>%  
  pivot_longer(  
    cols = G0.05:U0.3,  
    names_to = "sample",  
    values_to = "expression",  
    values_drop_na = TRUE  
  )
```

Enforcing one column per variable and one value per cell

Exercise 3

```
brauer4 <- brauer3 %>%  
  separate(  
    col = sample,  
    into = combine("nutrient", "rate"),  
    sep = 1  
  )
```

```
## Warning: `combine()` was deprecated in dplyr 1.0.0.  
## i Please use `vctrs::vec_c()` instead.  
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was  
## generated.
```

Exercise 4

```
brauer5 <- brauer4 %>%  
  separate(  
    col = NAME,  
    into = combine("gene_name", "biological_process", "molecular_function", "systematic_id", "n  
    sep = "\\|\\|\\|\\|"  
  )
```

```
## Warning: `combine()` was deprecated in dplyr 1.0.0.  
## i Please use `vctrs::vec_c()` instead.  
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was  
## generated.
```

Exercise 5

```
brauer_tidy <- brauer5 %>%  
  mutate_at(vars(gene_name:systematic_id), str_trim)
```

Visualizations using the tidy dataset

Exercise 6

```
brauer_tidy %>% filter(gene_name == "LEU1") %>%  
  ggplot()+  
  geom_line(aes(x=rate, y=expression, group = nutrient, color= nutrient ))
```

