CATEGORICAL DATA IN THE TIDYVERSE

Examining common themed variables

Emily Robinson
Data Scientist

Tidying data

```
  WorkChallengeFrequencyExplaining WorkChallengeFrequencyIntegration
  <chr>                            <chr>
1 Often                            Often
2 Most of the time                 Most of the time
```

```
  work_challenge frequency
  <chr>          <chr>
1 Explaining     Often
2 Explaining     Most of the time
3 Integration    Often
4 Integration    Most of the time
```

Selecting and gathering data

```
multipleChoiceResponses %>%
  select(contains("WorkChallengeFrequency")) %>%
  gather(work_challenge, frequency)
```

```
# A tibble: 367,752 x 2
   work_challenge                   frequency
   <chr>                            <chr>
 1 WorkChallengeFrequencyPolitics Rarely
 2 WorkChallengeFrequencyPolitics NA
 3 WorkChallengeFrequencyPolitics NA
 4 WorkChallengeFrequencyPolitics Often
 5 WorkChallengeFrequencyPolitics Often
 6 WorkChallengeFrequencyPolitics NA
 7 WorkChallengeFrequencyPolitics NA
 8 WorkChallengeFrequencyPolitics NA
```

Changing strings

```
work_challenges <- multipleChoiceResponses %>%
  select(contains("WorkChallengeFrequency")) %>%
  gather(work_challenge, frequency) %>%
  mutate(work_challenge = str_remove(work_challenge,
  "WorkChallengeFrequency"))
```

```
# A tibble: 367,752 x 2
   work_challenge frequency
   <chr>          <chr>
 1 Politics       Rarely
 2 Politics       NA
 3 Politics       NA
 4 Politics       Often
 5 Politics       Often
 6 Politics       NA
 7 Politics       NA
```

if_else() and summarizing

```
work_challenges %>%
   filter(!is.na(frequency))
   mutate(frequency = if_else(
            frequency %in% c("Most of the time", "Often"),
            1,
            0)
            ) %>%
  group_by(work_challenge) %>%
  summarise(perc_problem = mean(frequency))
```

```
# A tibble: 22 x 2
   work_challenge  perc_problem
   <chr>                  <dbl>
 1 Clarity               0.0930
 2 DataAccess            0.0923
 3 DataFunds             0.0367
 4 Deployment            0.0265
 5 DirtyData             0.176
 6 DomainExpertise       0.0573
```

CATEGORICAL DATA IN THE TIDYVERSE

Let's practice!

CATEGORICAL DATA IN THE TIDYVERSE

Tricks of ggplot2

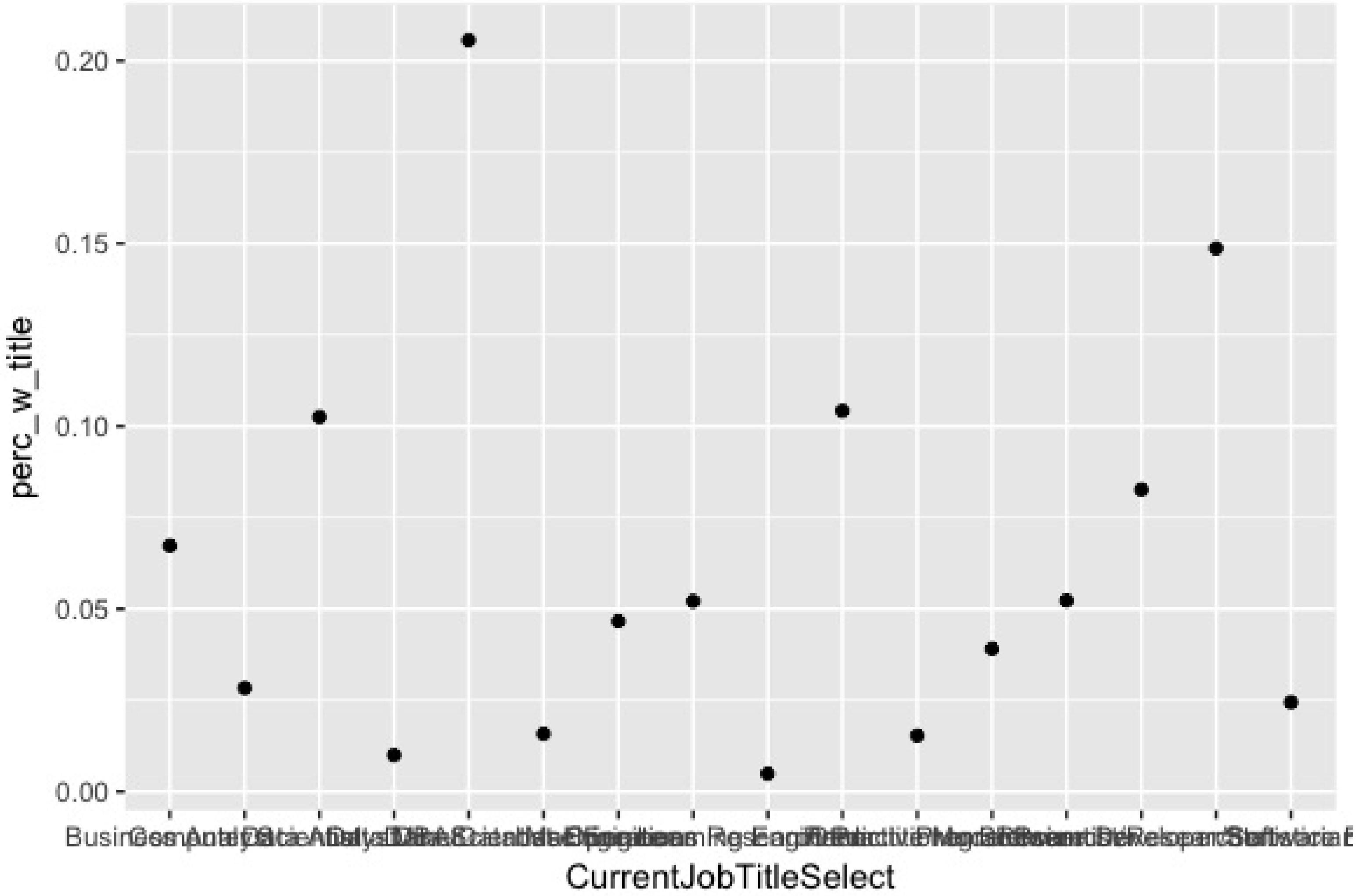
Emily Robinson
Instructor

Job title data

```
job_titles_by_perc
# A tibble: 16 x 2
   CurrentJobTitleSelect                perc_w_title
   <chr>                                       <dbl>
 1 Business Analyst                          0.0673
 2 Computer Scientist                        0.0283
 3 Data Analyst                              0.103
 4 Data Miner                                0.00997
 5 Data Scientist                            0.206
 6 DBA/Database Engineer                     0.0158
```

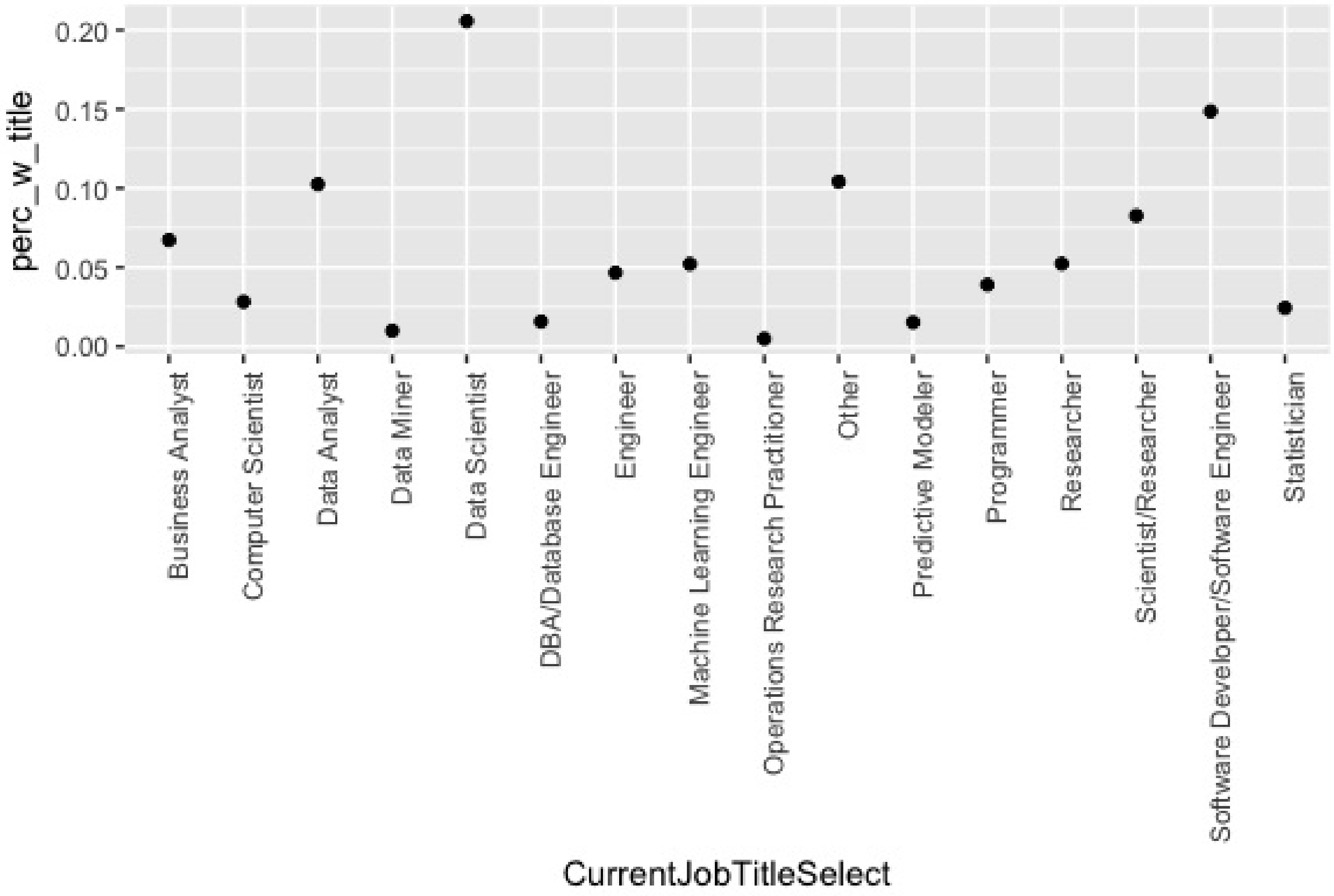
Initial plot

```
ggplot(job_titles_by_perc,
        aes(x = CurrentJobTitleSelect,, y = perc_w_title)) +
    geom_point()
```



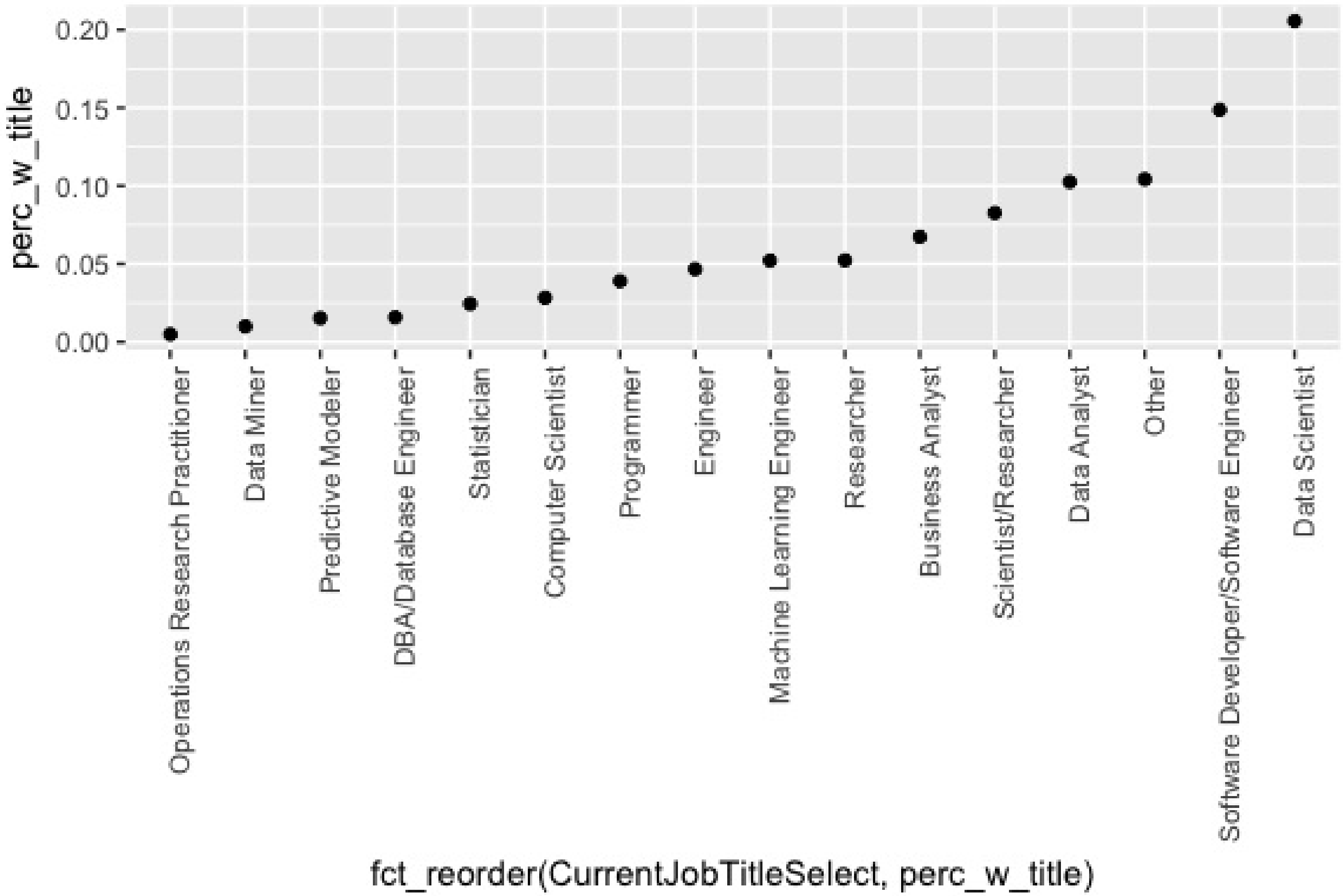
Changing tick labels angle

```
ggplot(job_titles_by_perc,
        aes(x = CurrentJobTitleSelect, y = perc_w_title)) +
    geom_point() +
    theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



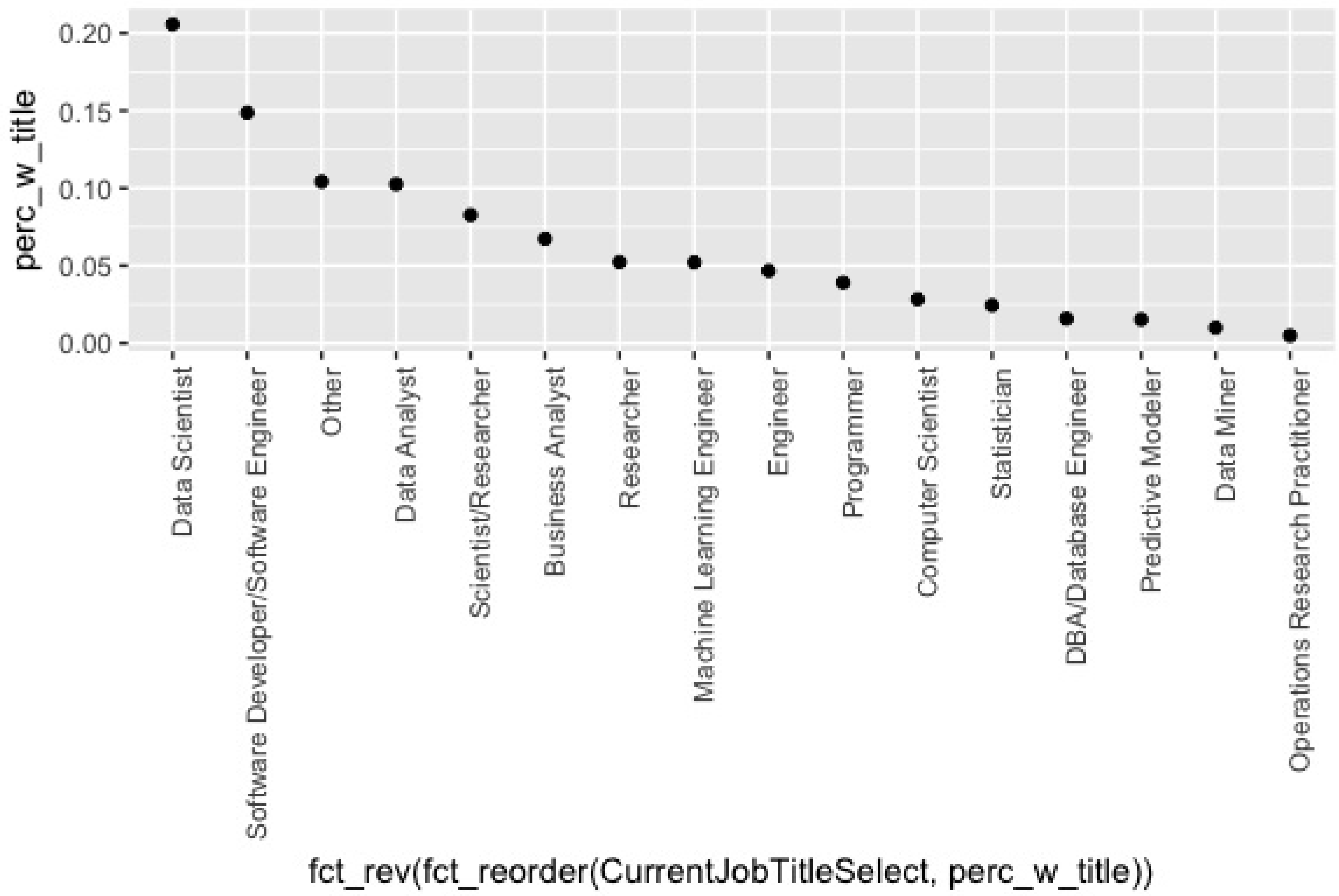
Using fct_reorder()

```
ggplot(job_titles_by_perc, aes(x = fct_reorder(CurrentJobTitleSelect,
                                perc_w_title), y = perc_w_title)) +
    geom_point() +
    theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



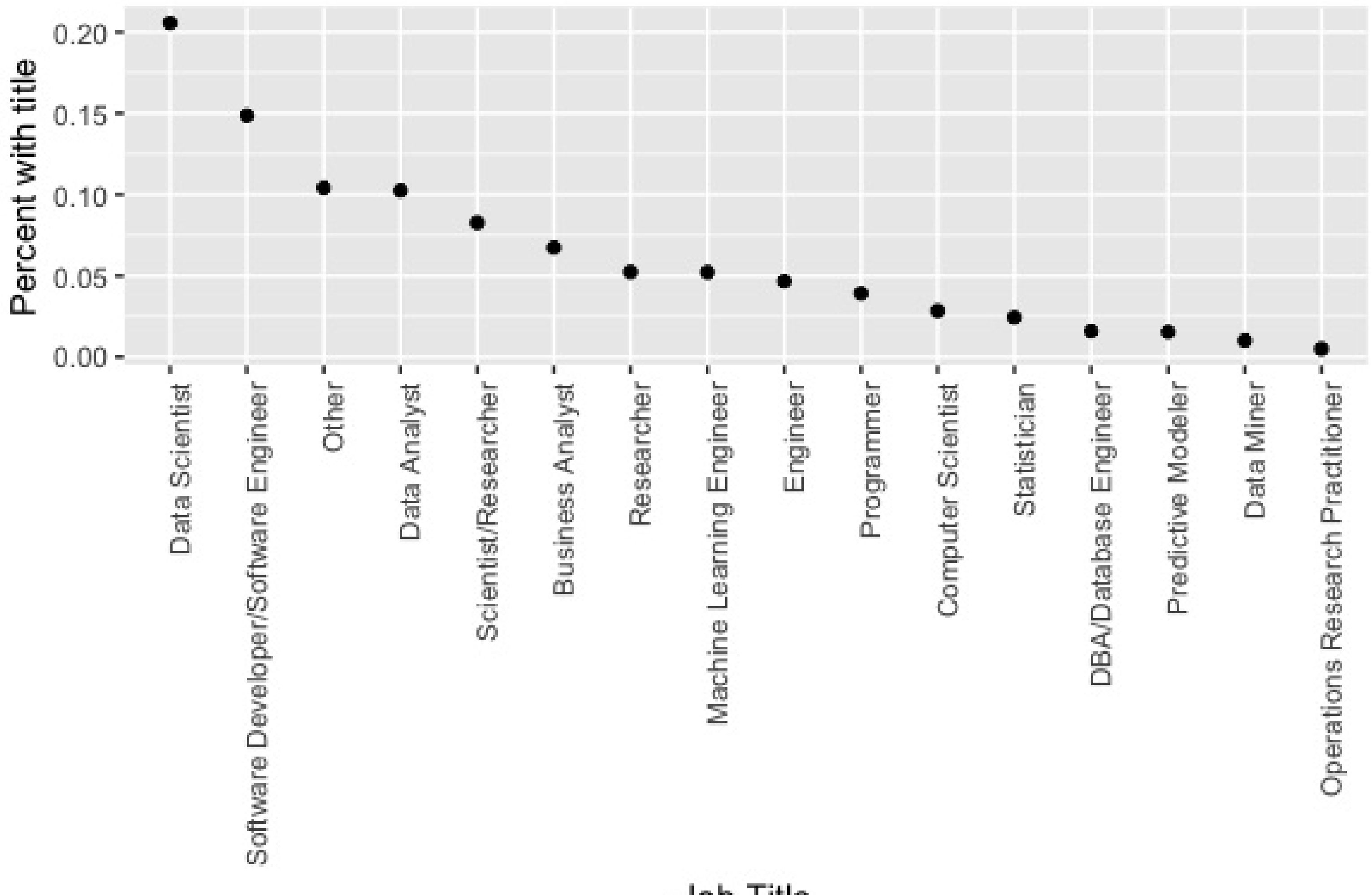
Adding fct_rev()

```
ggplot(job_titles_by_perc,
        aes(x = fct_rev(fct_reorder(CurrentJobTitleSelect, perc_w_title)),
            y = perc_w_title)) + geom_point() +
    theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



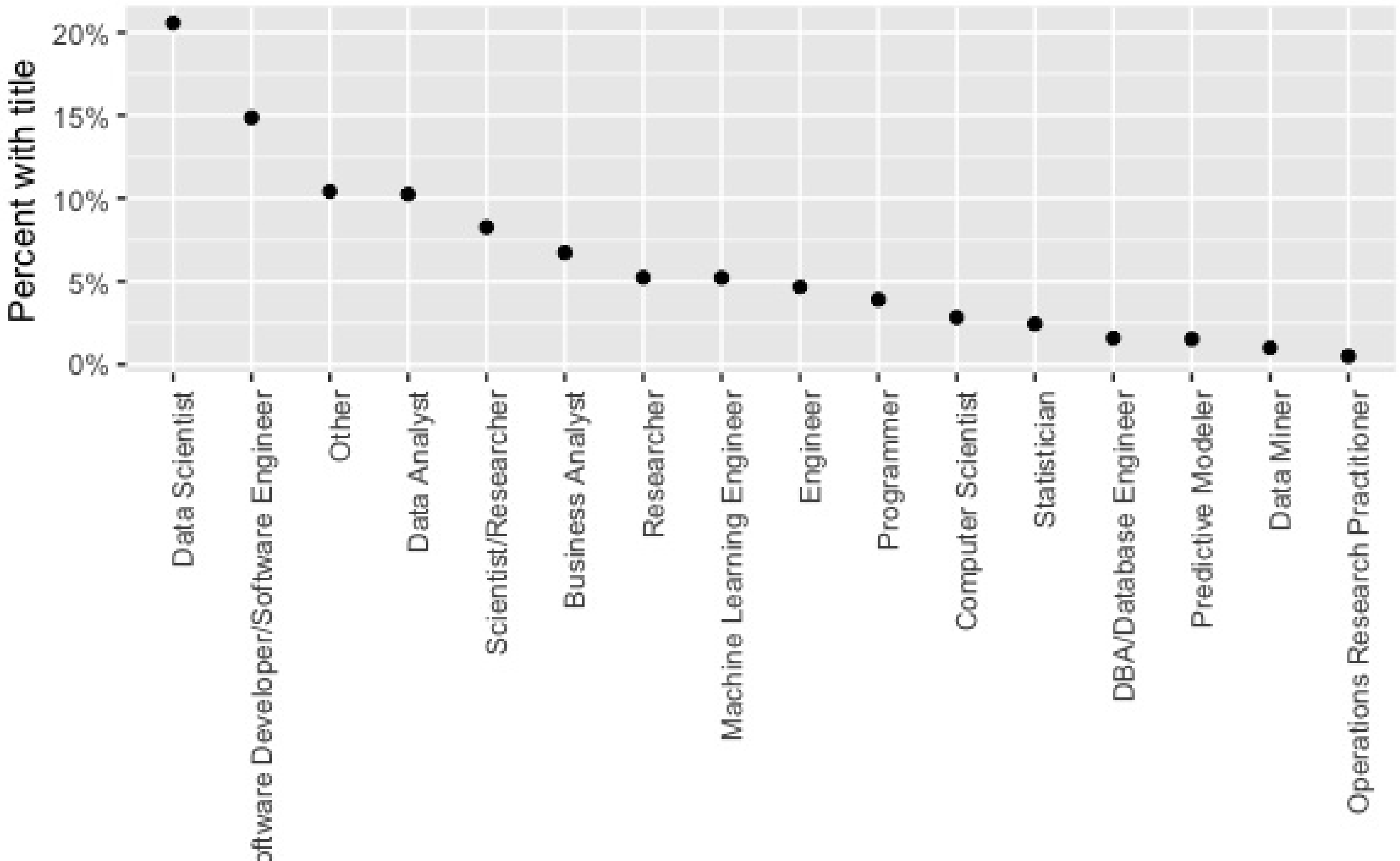
Using labs()

```
ggplot(job_titles_by_perc,
        aes(x = fct_rev(fct_reorder(CurrentJobTitleSelect, perc_w_title)),
            y = perc_w_title)) +
    geom_point() +
    theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
    labs(x = "Job Title", y = "Percent with title")
```



Changing to % scales

```
ggplot(job_titles_by_perc,
        aes(x = fct_rev(fct_reorder(CurrentJobTitleSelect, perc_w_title)),
        y = perc_w_title)) +
    geom_point() +
    theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
    labs(x = "Job Title", y = "Percent with title") +
    scale_y_continuous(labels = scales::percent_format())
```



CATEGORICAL DATA IN THE TIDYVERSE

Let's practice!

CATEGORICAL DATA IN THE TIDYVERSE

Changing and creating variables with case_when()

Emily Robinson
Data Scientist

case_when()

```
x <- 1:20
x
[1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18
[19] 19 20
```

```
case_when(x %% 15 == 0 ~ "fizz buzz",
          x %% 3 == 0 ~ "fizz",
          x %% 5 == 0 ~ "buzz",
          TRUE ~ as.character(x) )
```

```
 [1] "1"         "2"         "fizz"      "4"
 [5] "buzz"      "fizz"      "7"         "8"
 [9] "fizz"      "buzz"      "11"        "fizz"
[13] "13"        "14"        "fizz buzz" "16"
[17] "17"        "fizz"      "19"        "buzz"
```

Order matters

```
case_when(x %% 3 == 0 ~ "fizz buzz",
          x %% 5 == 0 ~ "buzz",
          x %% 3 == 0 ~ "fuzzy buzz",
          TRUE ~ as.character(x) )
```

```
 [1] "1"         "2"         "fizz buzz" "4"
 [5] "buzz"      "fizz buzz" "7"         "8"
 [9] "fizz buzz" "buzz"      "11"        "fizz buzz"
[13] "13"        "14"        "fizz buzz" "16"
[17] "17"        "fizz buzz" "19"        "buzz"
```

case_when() with multiple variables

```
> moods
# A tibble: 4 x 2
  mood  status
  <chr> <chr>
1 happy know it
2 happy do not know it
3 sad   know it
4 happy know it
```

```
moods %>%
  mutate(action = case_when(
  mood == "happy" & status == "know it" ~ "clap your hands",
   mood == "happy" & status == "do not know it" ~ "stomp your feet",
   mood == "sad" ~ "look at puppies",
   TRUE ~ "jump around")
```

CATEGORICAL DATA IN THE TIDYVERSE

Let's practice!