HR_analytics

Jiao Lai

4/4/2021

Steps to analyze HR data: 1. Identify groups to compare. 2. Calculate summary statistics for each group. 3. Compare the differences statistically or visually.

Case overview: 1. Identifying the best recruiting source. Quality of hire: retention, or how long the employee stays manager's satisfaction with the hire job performance amount of time it takes to become fully productive 2. What is driving low employee engagement? 3. Are new hires getting paid too much? 4. Are performance ratings being given consistently? 5. Improving employee safety with data.

Load data from website.

```
library(readr)
library(broom)
survey <- read_csv(url("https://assets.datacamp.com/production/course_5977/datasets/survey_data.csv"))
```

```
## Parsed with column specification:
## cols(
##   employee_id = col_double(),
##   department = col_character(),
##   engagement = col_double(),
##   salary = col_double(),
##   vacation_days_taken = col_double()
## )
```

```
recruitment <- read_csv(url("https://assets.datacamp.com/production/course_5977/datasets/recruitment_da
```

```
## Parsed with column specification:
## cols(
##   attrition = col_double(),
##   performance_rating = col_double(),
##   sales_quota_pct = col_double(),
##   recruiting_source = col_character()
## )
```

```
pay <- read_csv(url("https://assets.datacamp.com/production/course_5977/datasets/fair_pay_data.csv"))
```

```
## Parsed with column specification:
## cols(
##   employee_id = col_double(),
##   department = col_character(),
##   salary = col_double(),
##   new_hire = col_character(),
##   job_level = col_character()
## )
```

```
performance <- read_csv(url("https://assets.datacamp.com/production/course_5977/datasets/performance_da
```

```
## Parsed with column specification:
## cols(
##   employee_id = col_double(),
##   rating = col_double()
## )
```

```
hr_1 <- read_csv(url("https://assets.datacamp.com/production/course_5977/datasets/hr_data.csv"))
```

```
## Parsed with column specification:
## cols(
##   employee_id = col_double(),
##   department = col_character(),
##   job_level = col_character(),
##   gender = col_character()
## )
```

```
accident <- read_csv(url("https://assets.datacamp.com/production/course_5977/datasets/accident_data.csv
```

```
## Parsed with column specification:
## cols(
##   year = col_double(),
##   employee_id = col_double(),
##   accident_type = col_character()
## )
```

```
hr_2 <- read_csv(url("https://assets.datacamp.com/production/course_5977/datasets/hr_data_2.csv"))
```

```
## Parsed with column specification:
## cols(
##   year = col_double(),
##   employee_id = col_double(),
##   location = col_character(),
##   overtime_hours = col_double()
## )
```

```
survey_2 <- read_csv(url("https://assets.datacamp.com/production/course_5977/datasets/survey_data_2.csv
```

```
## Parsed with column specification:
## cols(
##   year = col_double(),
##   employee_id = col_double(),
##   engagement = col_double()
## )
```

The dataset 'recruitment' contains sources of recruiting, and three measurements of quality of hires. For this dataset, we are interested in whether quality of hires are different in terms of recruiting source.

```
head(recruitment)
```

```
## # A tibble: 6 x 4
##   attrition performance_rating sales_quota_pct recruiting_source
##       <dbl>              <dbl>           <dbl> <chr>
## 1         1                  3           1.09  Applied Online
## 2         0                  3           2.39  <NA>
## 3         1                  2           0.498 Campus
## 4         0                  2           2.51  <NA>
## 5         0                  3           1.42  Applied Online
## 6         1                  3           0.548 Referral
```

```r
names(recruitment)
```

```
## [1] "attrition"          "performance_rating" "sales_quota_pct"
## [4] "recruiting_source"
```

```r
summary(recruitment)
```

```
##    attrition     performance_rating sales_quota_pct   recruiting_source
##  Min.   :0.000   Min.   :1.000      Min.   :-0.7108   Length:446
##  1st Qu.:0.000   1st Qu.:2.000      1st Qu.: 0.5844   Class :character
##  Median :0.000   Median :3.000      Median : 1.0701   Mode  :character
##  Mean   :0.213   Mean   :2.895      Mean   : 1.0826
##  3rd Qu.:0.000   3rd Qu.:3.000      3rd Qu.: 1.5325
##  Max.   :1.000   Max.   :5.000      Max.   : 3.6667
```

```r
colSums(is.na(recruitment))
```

```
##          attrition performance_rating    sales_quota_pct  recruiting_source
##                  0                  0                  0                205
```

```r
levels(recruitment$recruiting_source)
```

```
## NULL
```

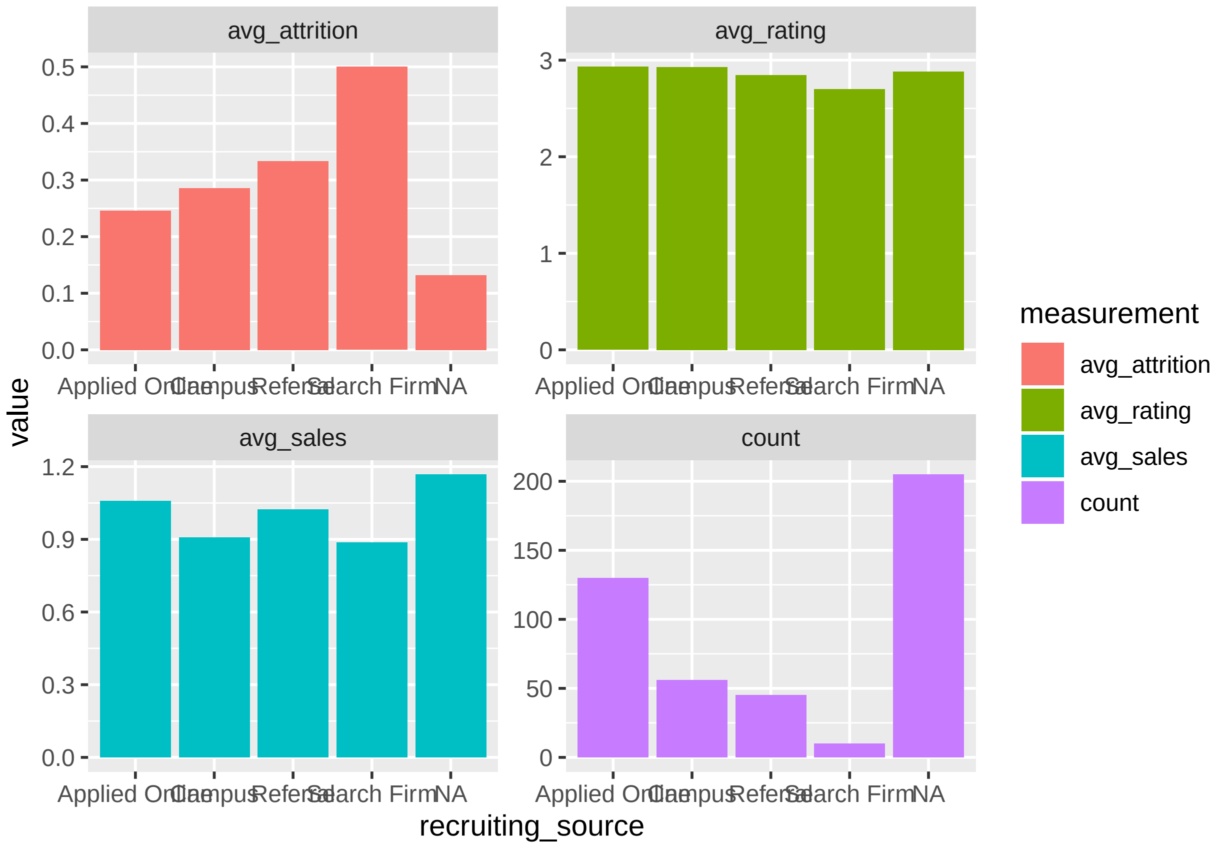
```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(tidyr)
recruitment_summary <- recruitment %>%
  group_by(recruiting_source) %>%
  summarize(
    count = n(),
    avg_attrition = mean(attrition),
    avg_rating = mean(performance_rating),
    avg_sales = mean(sales_quota_pct),
  ) %>%
  gather( "measurement", 'value', -recruiting_source)
library(ggplot2)
recruitment_summary %>%
  ggplot(aes(x = recruiting_source, y = value, fill = measurement))+
  geom_col(position = 'dodge') +
  facet_wrap(~measurement, scales = 'free')
```



The dataset 'survey' contains info for employees: salary, department, engagement, vacation days taken.

```
head(survey)
```

```
## # A tibble: 6 x 5
##   employee_id department  engagement  salary vacation_days_taken
##         <dbl> <chr>            <dbl>   <dbl>               <dbl>
## 1           1 Sales                3 103264.                   7
## 2           2 Engineering          3  80709.                  12
## 3           4 Engineering          3  60737.                  12
## 4           5 Engineering          3  99116.                   7
## 5           7 Engineering          3  51022.                  18
## 6           8 Engineering          3  98400.                   9
```

```
summary(survey)
```

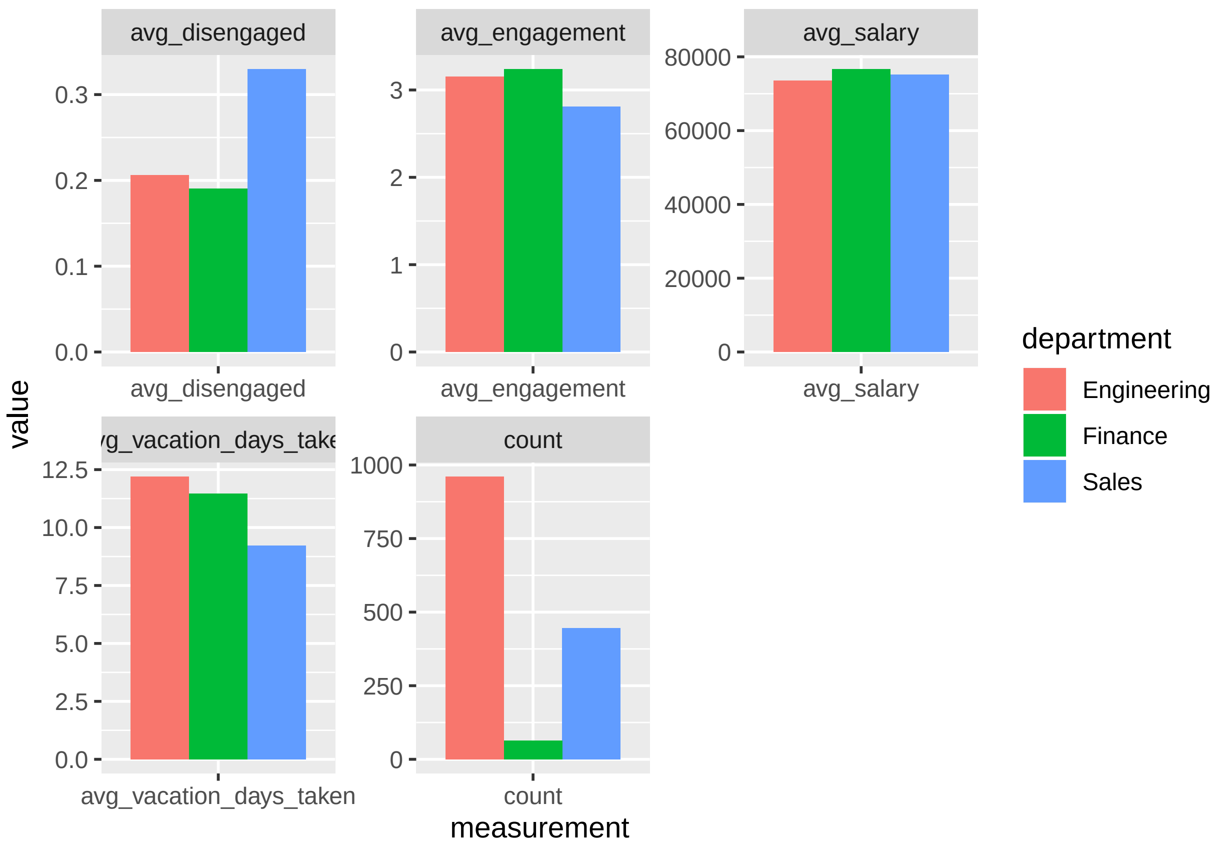
```
##   employee_id      department          engagement       salary
##  Min.   :   1.0   Length:1470        Min.   :1.00   Min.   : 45530
##  1st Qu.: 491.2   Class :character   1st Qu.:3.00   1st Qu.: 59407
##  Median :1020.5   Mode  :character   Median :3.00   Median : 70481
##  Mean   :1024.9                      Mean   :3.05   Mean   : 74162
##  3rd Qu.:1555.8                      3rd Qu.:4.00   3rd Qu.: 84763
##  Max.   :2068.0                      Max.   :5.00   Max.   :164073
##  vacation_days_taken
##  Min.   : 0.00
##  1st Qu.: 6.00
##  Median :10.00
##  Mean   :11.27
##  3rd Qu.:16.00
```

```
## Max.   :38.00
```

```
unique(survey$department)
```

```
## [1] "Sales"       "Engineering" "Finance"
```

```
survey_summary <- survey %>%
  mutate(disengaged = ifelse(engagement %in% c(1, 2), 1, 0)) %>%
  group_by(department) %>%
  summarize(
    count = n(),
    avg_engagement = mean(engagement),
    avg_salary = mean(salary),
    avg_vacation_days_taken = mean(vacation_days_taken),
    avg_disengaged = mean(disengaged)
  )
survey_summary %>%
  gather("measurement", "value", -department) %>%
  ggplot(aes(x = measurement, y = value, fill = department)) +
  geom_col(position = 'dodge') +
  facet_wrap(~measurement, scales = 'free')
```



To test if two groups are statistically significant different from each other, we can use t-test if the variable we are comparing is continuous, and chi-square test if the variable we are comparing is categorical.

```
survey <- survey %>%
  mutate(in_sales = ifelse(department == 'Sales', "Sales", "Other"),
         disengaged = ifelse(engagement %in% c(1,2), 1, 0))
## check if Sales and other department have different 'disengaged' and 'vacation_days_taken'
chisq.test(survey$in_sales, survey$disengaged)
```

```
## 
##  Pearson's Chi-squared test with Yates' continuity correction
## 
## data:  survey$in_sales and survey$disengaged
## X-squared = 25.524, df = 1, p-value = 4.368e-07
```

```r
## check if the two groups have equal variance
var.test(vacation_days_taken ~ in_sales, survey)
```

```
## 
##  F test to compare two variances
## 
## data:  vacation_days_taken by in_sales
## F = 1.4908, num df = 1023, denom df = 445, p-value = 1.435e-06
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  1.270286 1.740838
## sample estimates:
## ratio of variances 
##           1.490751
```

```r
## check if the two groups are normal
with(survey, shapiro.test(vacation_days_taken[in_sales == 'Sales']))
```

```
## 
##  Shapiro-Wilk normality test
## 
## data:  vacation_days_taken[in_sales == "Sales"]
## W = 0.9415, p-value = 3.004e-12
```

```r
with(survey, shapiro.test(vacation_days_taken[in_sales == 'Other']))
```

```
## 
##  Shapiro-Wilk normality test
## 
## data:  vacation_days_taken[in_sales == "Other"]
## W = 0.96065, p-value = 5.534e-16
```

```r
t.test(vacation_days_taken ~ in_sales, survey, var.equal = FALSE)
```

```
## 
##  Welch Two Sample t-test
## 
## data:  vacation_days_taken by in_sales
## t = 8.1549, df = 1022.9, p-value = 1.016e-15
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  2.229473 3.642409
## sample estimates:
## mean in group Other mean in group Sales 
##           12.160156            9.224215
```

The table 'pay' contains employee_id, department, salary, whether they are new_hire, and their job_level.

```r
names(pay)
```

```
## [1] "employee_id" "department"  "salary"      "new_hire"    "job_level"
```

```r
head(pay)
```

```
## # A tibble: 6 x 5
##   employee_id department   salary new_hire job_level
##         <dbl> <chr>         <dbl> <chr>    <chr>
## 1           1 Sales       103264. No       Salaried
## 2           2 Engineering  80709. No       Hourly
## 3           4 Engineering  60737. Yes      Hourly
## 4           5 Engineering  99116. Yes      Salaried
## 5           7 Engineering  51022. No       Hourly
## 6           8 Engineering  98400. No       Salaried
```

```r
summary(pay)
```

```
##   employee_id      department            salary         new_hire
##  Min.   :   1.0   Length:1470        Min.   : 43820   Length:1470
##  1st Qu.: 491.2   Class :character   1st Qu.: 59378   Class :character
##  Median :1020.5   Mode  :character   Median : 70425   Mode  :character
##  Mean   :1024.9                      Mean   : 74142
##  3rd Qu.:1555.8                      3rd Qu.: 84809
##  Max.   :2068.0                      Max.   :164073
##   job_level
##  Length:1470
##  Class :character
##  Mode  :character
##
##
##
```

```r
pay %>%
  group_by(new_hire) %>%
  summarize(
    count = n(),
    avg_salary = mean(salary))
```

```
## # A tibble: 2 x 3
##   new_hire count avg_salary
##   <chr>    <int>      <dbl>
## 1 No        1072     73425.
## 2 Yes        398     76074.
```

```r
var.test(salary ~ new_hire, pay)
```

```
##
##  F test to compare two variances
##
## data:  salary by new_hire
## F = 0.9208, num df = 1071, denom df = 397, p-value = 0.3118
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.7798464 1.0805258
## sample estimates:
## ratio of variances
##           0.920799
```

```r
with(pay, shapiro.test(salary[new_hire == 'Yes']))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  salary[new_hire == "Yes"]
## W = 0.92883, p-value = 7.914e-13
```

```r
with(pay, shapiro.test(salary[new_hire == 'No']))
```

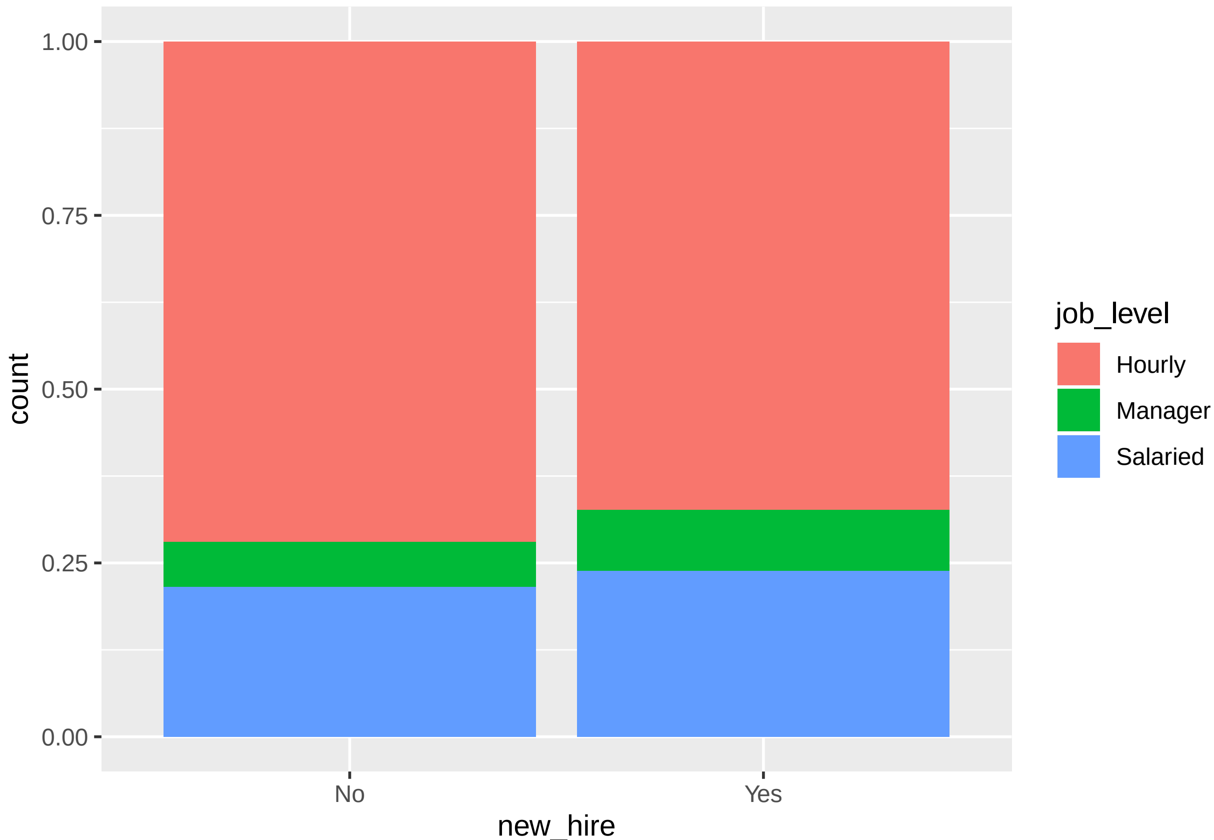
```
##
##  Shapiro-Wilk normality test
##
## data:  salary[new_hire == "No"]
## W = 0.93073, p-value < 2.2e-16
```

```r
t.test(salary ~ new_hire, pay, var.equal = TRUE)
```

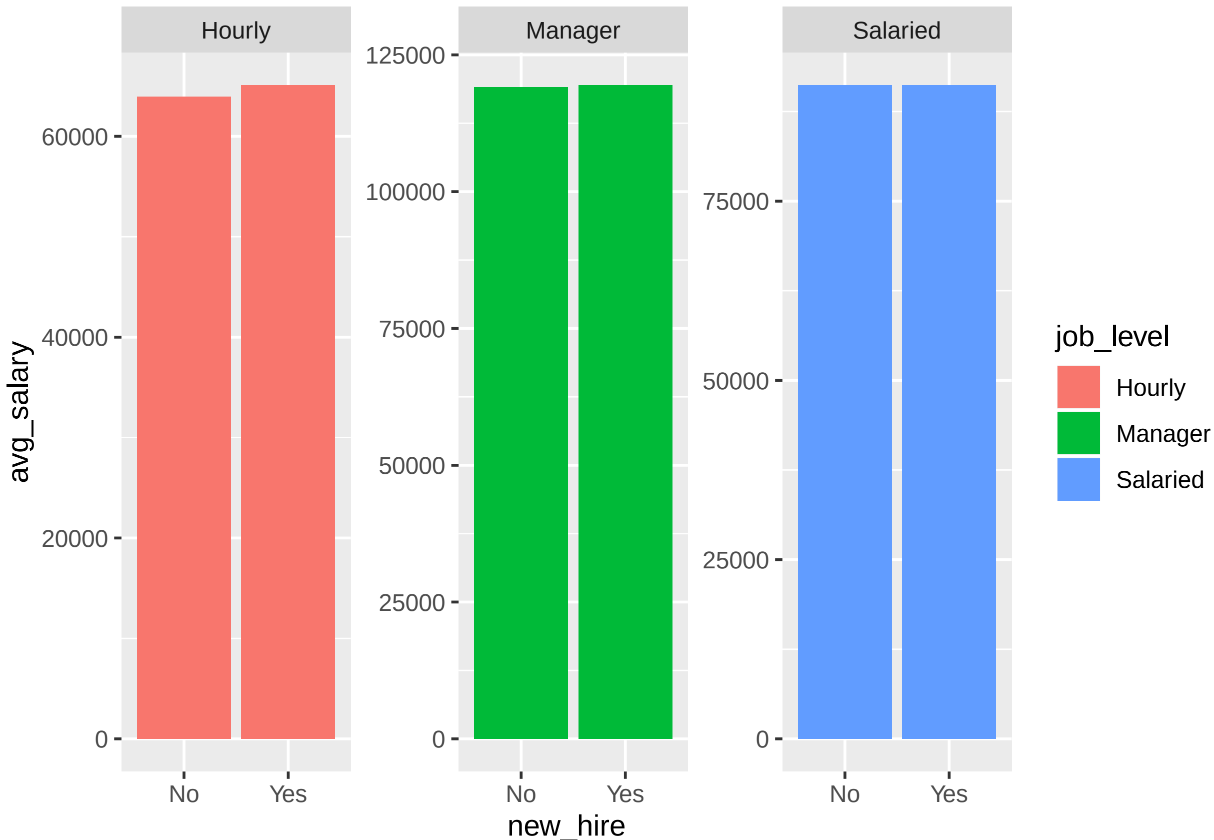
```
##
##  Two Sample t-test
##
## data:  salary by new_hire
## t = -2.3885, df = 1468, p-value = 0.01704
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -4825.7656  -473.5786
## sample estimates:
##  mean in group No mean in group Yes
##          73424.60          76074.28
```

Check for omitted variables.

```r
pay %>%
  ggplot(aes(x = new_hire, fill = job_level)) +
  geom_bar(position = 'fill')
```




```
pay %>%
  group_by(new_hire, job_level) %>%
  summarize(avg_salary = mean(salary)) %>%
  ggplot(aes(x = new_hire, y = avg_salary, fill = job_level)) +
  geom_col(position = 'dodge') +
  facet_wrap(~job_level, scales = 'free')
```



Look at 'hourly' job_level only.

```
hourly <- pay %>% filter(job_level == 'Hourly')
var.test(salary ~ new_hire, hourly)
```

```
##
##  F test to compare two variances
##
## data:  salary by new_hire
## F = 1.1701, num df = 770, denom df = 267, p-value = 0.1268
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.9561283 1.4183790
## sample estimates:
## ratio of variances
##           1.170114
```

```
tidy(t.test(salary ~ new_hire, hourly, var.equal = TRUE))
```

```
## # A tibble: 1 x 9
##   estimate1 estimate2 statistic p.value parameter conf.low conf.high method
##       <dbl>     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>     <dbl> <chr>
## 1    63966.    65073.     -1.69  0.0923      1037   -2396.      182. " Two...
## # ... with 1 more variable: alternative <chr>
```

Use linear regression to control confounding variables.

```
## Simple linear regression.
## Simple linear regression gives the same result with t test of equal variances.
lm.simple <- lm(salary ~ new_hire, pay) %>% tidy()
```

```r
## Add job_level.
lm.mul <- lm(salary ~ new_hire + job_level, pay) %>% tidy()
```

Analyze HR data from different resources.

```r
summary(hr_1)
```

```
##   employee_id      department          job_level            gender         
##  Min.   :   1.0   Length:1470        Length:1470        Length:1470       
##  1st Qu.: 491.2   Class :character   Class :character   Class :character  
##  Median :1020.5   Mode  :character   Mode  :character   Mode  :character  
##  Mean   :1024.9                                                           
##  3rd Qu.:1555.8                                                           
##  Max.   :2068.0
```

```r
summary(performance)
```

```
##   employee_id         rating    
##  Min.   :   1.0   Min.   :1.00  
##  1st Qu.: 491.2   1st Qu.:2.00  
##  Median :1020.5   Median :3.00  
##  Mean   :1024.9   Mean   :2.83  
##  3rd Qu.:1555.8   3rd Qu.:4.00  
##  Max.   :2068.0   Max.   :5.00
```

```r
joined <- hr_1 %>%
  left_join(performance, by = 'employee_id')
summary(joined)
```

```
##   employee_id      department          job_level            gender         
##  Min.   :   1.0   Length:1470        Length:1470        Length:1470       
##  1st Qu.: 491.2   Class :character   Class :character   Class :character  
##  Median :1020.5   Mode  :character   Mode  :character   Mode  :character  
##  Mean   :1024.9                                                           
##  3rd Qu.:1555.8                                                           
##  Max.   :2068.0                                                           
##      rating    
##  Min.   :1.00  
##  1st Qu.:2.00  
##  Median :3.00  
##  Mean   :2.83  
##  3rd Qu.:4.00  
##  Max.   :5.00
```

```r
joined %>%
  group_by(gender) %>%
  summarize(avg_rating = mean(rating, na.rm = TRUE))
```

```
## # A tibble: 2 x 2
##   gender avg_rating
##   <chr>       <dbl>
## 1 Female       2.75
## 2 Male         2.92
```

Compare performance by gender.

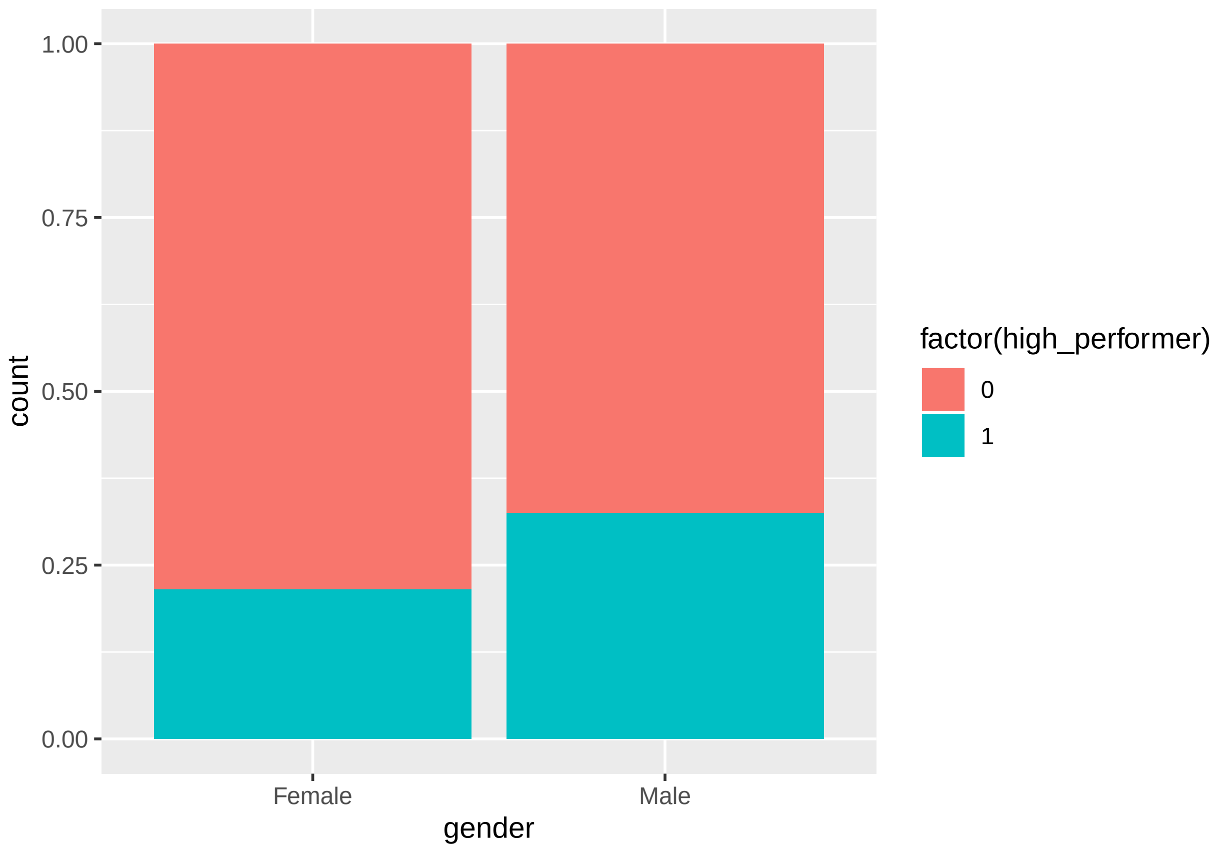
```r
joined <- joined %>%
  mutate(high_performer = ifelse(rating >= 4, 1, 0))
## Compare the difference in performance by gender
chisq.test(joined$high_performer, joined$gender)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  joined$high_performer and joined$gender
## X-squared = 22.229, df = 1, p-value = 2.42e-06
```

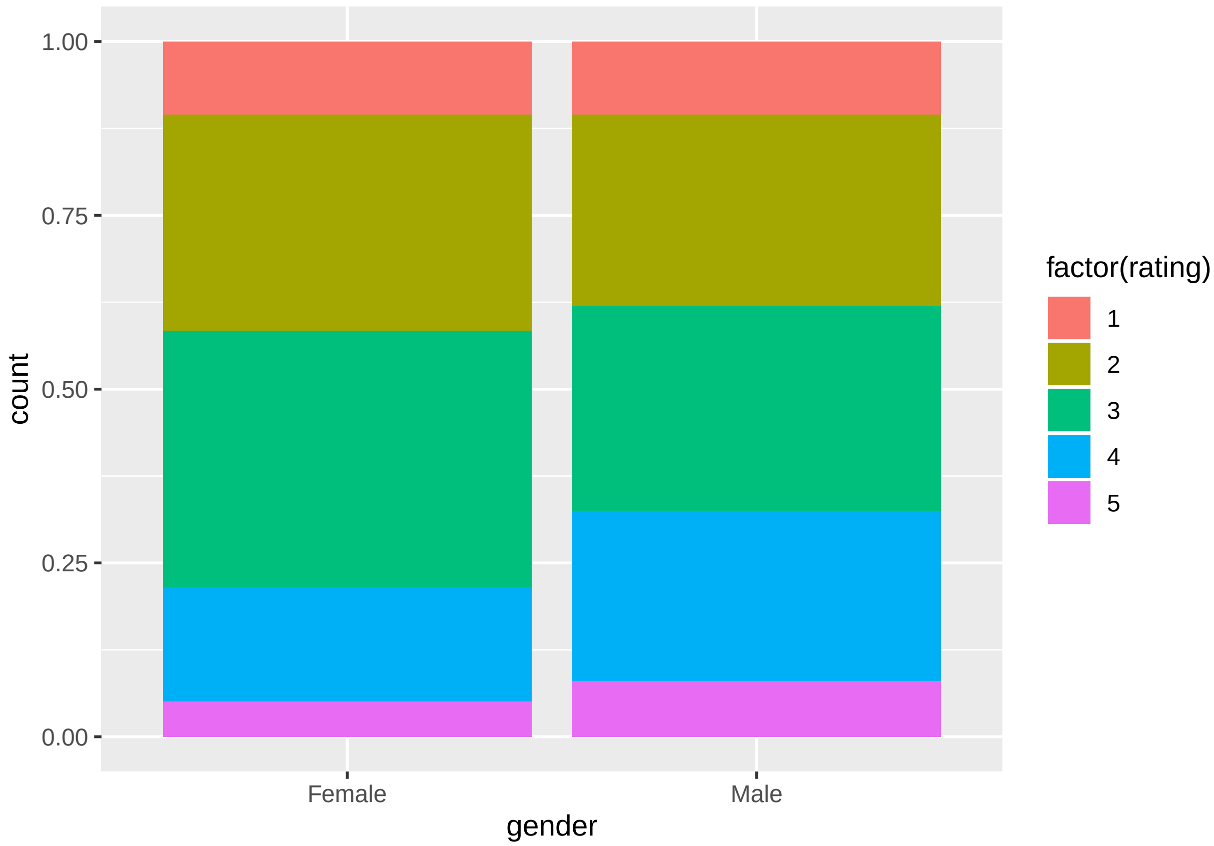
```r
chisq.test(joined$rating, joined$gender)
```

```
##
##  Pearson's Chi-squared test
##
## data:  joined$rating and joined$gender
## X-squared = 24.501, df = 4, p-value = 6.336e-05
```

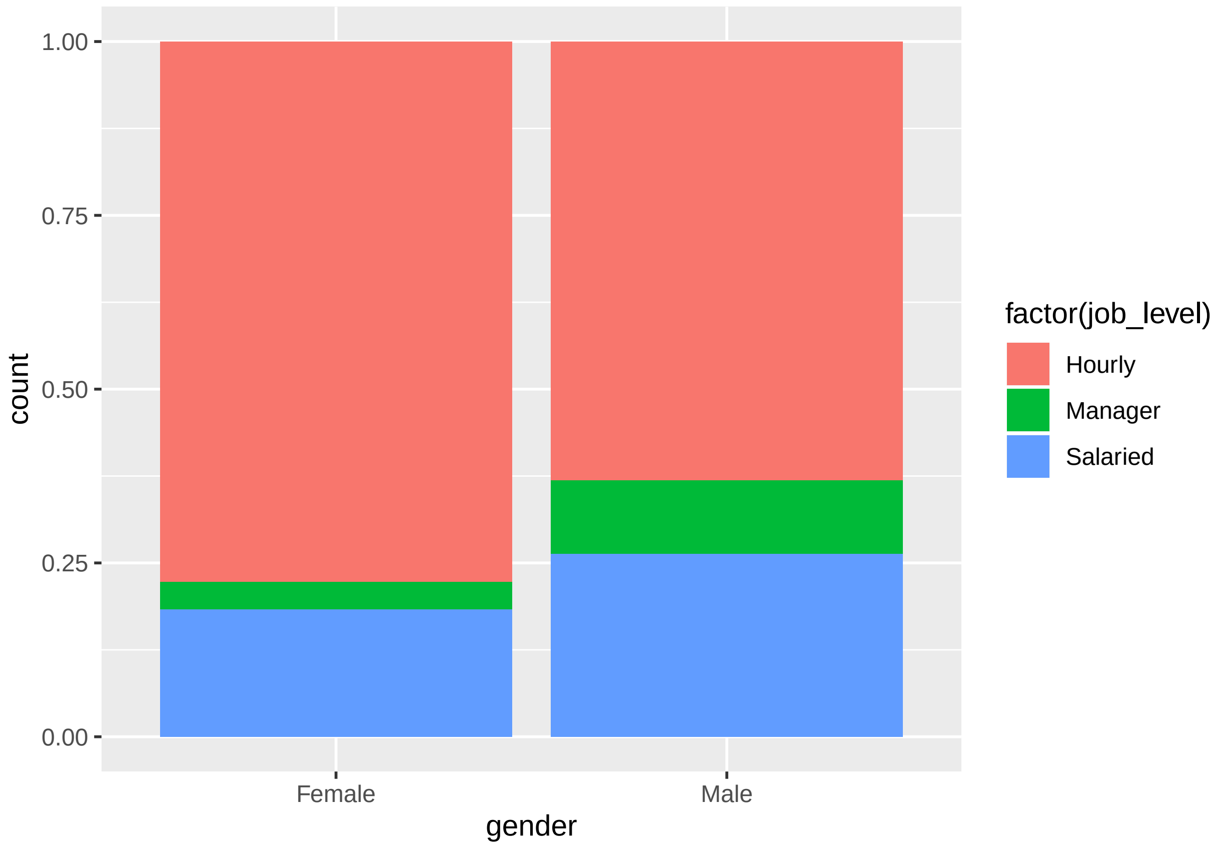
```r
## visualize the distribution of performance by gender
joined %>%
  ggplot(aes(x = gender, fill = factor(high_performer)))+
  geom_bar(position = 'fill')
```




```r
joined %>%
  ggplot(aes(x = gender, fill = factor(rating)))+
  geom_bar(position = 'fill')
```



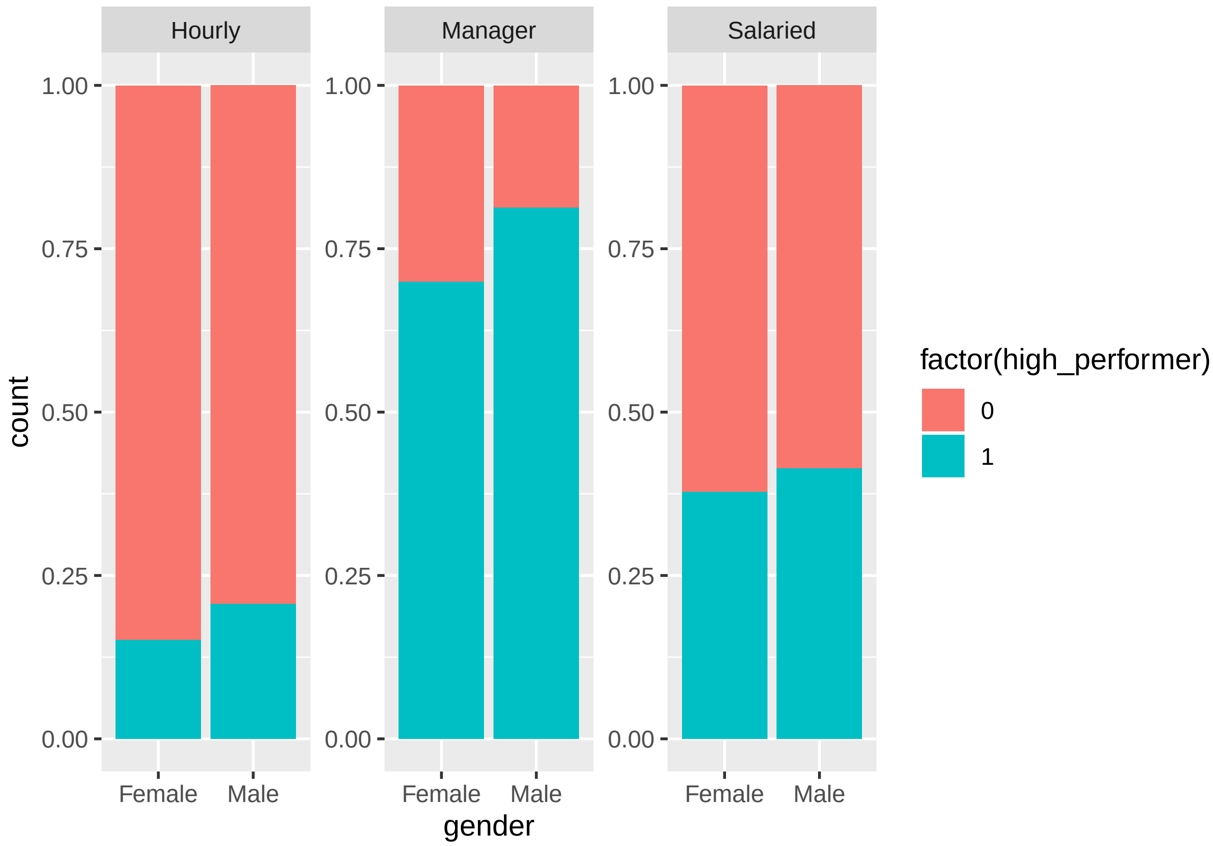

```r
joined %>%
  ggplot(aes(x = gender, fill = factor(job_level)))+
  geom_bar(position = 'fill')
```




```
## check if job_level distribution is different by gender
chisq.test(joined$job_level, joined$gender)
```

```
##
##  Pearson's Chi-squared test
##
## data:  joined$job_level and joined$gender
## X-squared = 44.506, df = 2, p-value = 2.166e-10
```

```
## visualize the difference in performance by gender and job_level
joined %>%
  ggplot(aes(x = gender, fill = factor(high_performer)))+
  geom_bar(position = 'fill') +
  facet_wrap(~job_level, scales = 'free')
```



Use logistic regression to predict binary variable. Find variables affecting an employee's chance to be high_performer.

```
glm.simple <- glm(high_performer ~ gender, joined, family = 'binomial') %>% tidy()
glm.mul <- glm(high_performer ~ gender + job_level, joined, family = 'binomial')  %>% tidy()
```

Analyze workforce safety.

```
head(hr_2)
```

```
## # A tibble: 6 x 4
##    year employee_id location    overtime_hours
##   <dbl>       <dbl> <chr>                <dbl>
## 1  2016           1 Northwood               14
## 2  2017           1 Northwood                8
## 3  2016           2 East Valley              8
## 4  2017           2 East Valley             11
## 5  2016           4 East Valley              4
## 6  2017           4 East Valley              2
```

```
head(accident)
```

```
## # A tibble: 6 x 3
##    year employee_id accident_type
##   <dbl>       <dbl> <chr>
## 1  2017           1 Mild
## 2  2017           4 Mild
## 3  2017          11 Mild
## 4  2017          19 Mild
## 5  2017          22 Mild
```

```
## 6  2016          23 Mild
```

```
acc_joined <- hr_2 %>%
  left_join(accident, by = c('employee_id', 'year')) %>%
  mutate(had_accident = ifelse(is.na(accident_type), 0, 1))
## Accident rate by year.
acc_joined %>%
  group_by(year) %>%
  summarize(avg_accident = mean(had_accident))
```

```
## # A tibble: 2 x 2
##    year avg_accident
##   <dbl>        <dbl>
## 1  2016       0.0850
## 2  2017       0.120
```

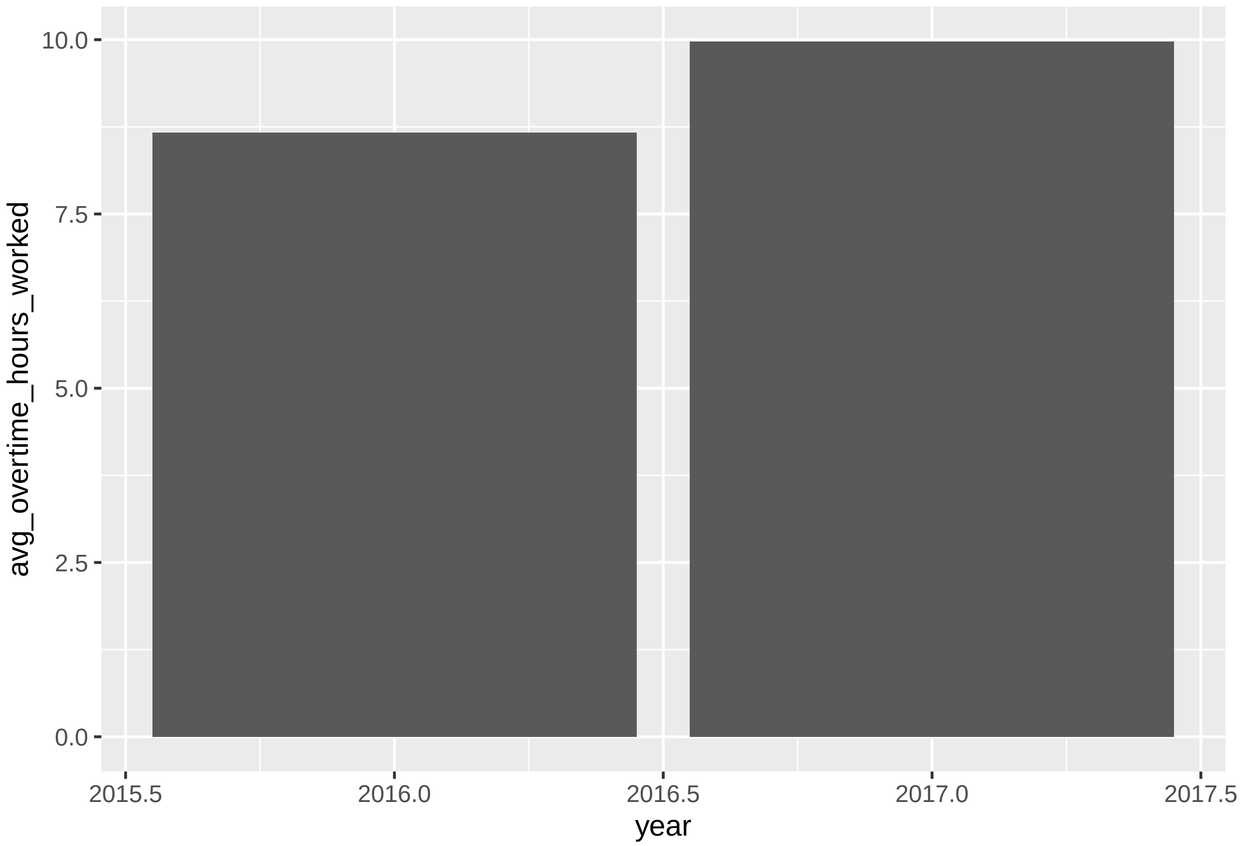
```
chisq.test(acc_joined$had_accident, acc_joined$year)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  acc_joined$had_accident and acc_joined$year
## X-squared = 9.5986, df = 1, p-value = 0.001947
```

```
acc_joined %>%
  group_by(year, location) %>%
  summarize(avg_accident = mean(had_accident)) %>%
  ggplot(aes(x = factor(year), y = avg_accident, fill = location)) +
  geom_col(position = 'dodge') +
  facet_wrap(~location, scales = 'free')
```

Looked at subset of data with interest: Southfield.

```
southfield <- acc_joined %>%
  filter(location == 'Southfield')
southfield %>%
  group_by(year) %>%
  summarize(avg_overtime_hours_worked = mean(overtime_hours)) %>%
  ggplot(aes(x = year, y = avg_overtime_hours_worked)) +
  geom_col()
```

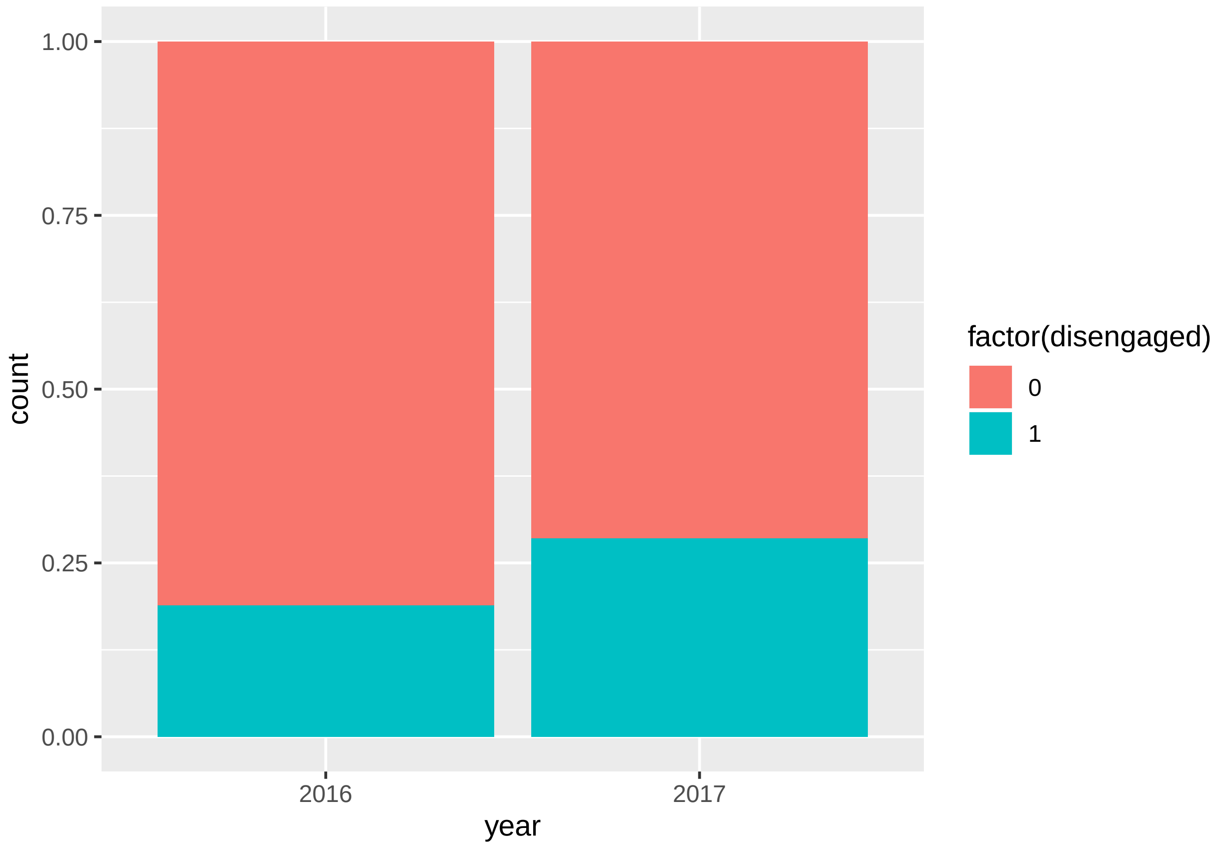



```
t.test(overtime_hours ~ year, southfield)
```

```
##
##  Welch Two Sample t-test
##
## data:  overtime_hours by year
## t = -1.6043, df = 595.46, p-value = 0.1092
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.904043  0.292747
## sample estimates:
## mean in group 2016 mean in group 2017
##           8.667774           9.973422
```

Use more data to check for sources of variation.

```
acc_survey <- southfield %>% left_join(survey_2, by = c('employee_id', 'year')) %>%
  mutate(disengaged = ifelse(engagement <= 2, 1, 0),
         year = as.factor(year))
acc_survey %>%
  ggplot(aes(x = year, fill = factor(disengaged))) +
  geom_bar(position = 'fill')
```




```r
chisq.test(acc_survey$disengaged, acc_survey$year)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  acc_survey$disengaged and acc_survey$year
## X-squared = 7.1906, df = 1, p-value = 0.007329
```

```r
## check accident trend in other fields.
other <- acc_joined %>% filter(location != 'Southfield')
other %>%
  group_by(year) %>%
  summarize(avg_accident = mean(had_accident))
```

```
## # A tibble: 2 x 2
##    year avg_accident
##   <dbl>        <dbl>
## 1  2016       0.0873
## 2  2017       0.118
```

```r
chisq.test(other$had_accident, other$year)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  other$had_accident and other$year
## X-squared = 5.6881, df = 1, p-value = 0.01708
```

Use regression to control other variables.

```r
glm(had_accident ~ year + disengaged, data = acc_survey, family = 'binomial') %>% tidy()
```

```
## # A tibble: 3 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)   -2.92      0.250    -11.7  1.74e-31
## 2 year2017       0.440     0.285      1.55 1.22e- 1
## 3 disengaged     1.44      0.278      5.19 2.13e- 7
```