Untitled

Musaab Farooqui

8/13/2021

2 Causality

2.1 Racial Discrimination in the Labor Market

```
data("resume", package = "qss")
dim(resume)
```

```
## [1] 4870    4
```

```
#dimensions of table (x,y)
summary(resume)
```

```
##   firstname             sex                race                call        
##  Length:4870        Length:4870        Length:4870        Min.   :0.00000  
##  Class :character   Class :character   Class :character   1st Qu.:0.00000  
##  Mode  :character   Mode  :character   Mode  :character   Median :0.00000  
##                                                           Mean   :0.08049  
##                                                           3rd Qu.:0.00000  
##                                                           Max.   :1.00000
```

```
#summarizes table by class, character, length, and elemntary stats.
head(resume)
```

```
##   firstname    sex  race call
## 1   Allison female white    0
## 2   Kristen female white    0
## 3   Lakisha female black    0
## 4   Latonya female black    0
## 5    Carrie female white    0
## 6       Jay   male white    0
```

```
#produces first six rows, add comma after resume to produce more.
```

```
glimpse(resume)
```

```
## Rows: 4,870
## Columns: 4
## $ firstname <chr> "Allison", "Kristen", "Lakisha", "Latonya", "Carrie", "Jay",~
## $ sex       <chr> "female", "female", "female", "female", "female", "male", "f~
## $ race      <chr> "white", "white", "black", "black", "white", "white", "white~
## $ call      <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
```

```
#names of columns, type of character, and gives numbers.
```

```
race_call_tab <- resume %>%
  group_by(race, call) %>%
  count() %>%
  ungroup()
race_call_tab
```

```
## # A tibble: 4 x 3
##   race   call     n
##   <chr> <int> <int>
## 1 black     0  2278
## 2 black     1   157
## 3 white     0  2200
## 4 white     1   235
```

```
race_call_rate <- race_call_tab %>%
  group_by(race) %>%
  mutate(call_rate = n/sum(n)) %>%
  filter(call == 1 ) %>%
  select(race, call_rate)
  race_call_rate
```

```
## # A tibble: 2 x 2
## # Groups:   race [2]
##   race  call_rate
##   <chr>     <dbl>
## 1 black    0.0645
## 2 white    0.0965
```

```
resume %>%
  summarise(call_back = mean(call))
```

```
##    call_back
## 1 0.08049281
```

2.2 Subsetting Data in R

```
resume %>%
  mutate(BlackFemale = if_else(race == "black" & sex == "female", 1, 0)) %>%
  group_by(BlackFemale, race, sex) %>%
  count()
```

```
## # A tibble: 4 x 4
## # Groups:   BlackFemale, race, sex [4]
##   BlackFemale race  sex        n
##         <dbl> <chr> <chr>  <int>
## 1           0 black male     549
## 2           0 white female  1860
## 3           0 white male     575
## 4           1 black female  1886
```

```
class(1)
```

```
## [1] "numeric"
```

```
class(1L)
```

```
## [1] "integer"
```

```
class(1:5)
```

```
## [1] "integer"
```

```
class(c(1,2,3))
```

```
## [1] "numeric"
```

```
class(as.integer(c(1,2,3)))
```

```
## [1] "integer"
```

2.2.3 Factor Variables

```
resume %>%
  mutate(
    race_sex = case_when(
      race == "black" & sex == "female" ~ "black female",
      race == "white" & sex == "female" ~ "white female",
      race == "black" & sex == "male" ~ "black male",
      race == "white" & sex == "male" ~ "white male"
    )
  ) %>%
  head()
```

```
##   firstname    sex  race call     race_sex
## 1   Allison female white    0 white female
## 2   Kristen female white    0 white female
## 3   Lakisha female black    0 black female
## 4   Latonya female black    0 black female
## 5    Carrie female white    0 white female
## 6       Jay   male white    0   white male
```

```
resume %>%
  mutate(
    race_sex = case_when(
      race == "black" & sex == "female" ~ "black female",
      race == "black" & sex == "male" ~ "black male",
      TRUE ~ "white"
    )
  ) %>%
  head()
```

```
##   firstname    sex  race call     race_sex
## 1   Allison female white    0        white
## 2   Kristen female white    0        white
## 3   Lakisha female black    0 black female
## 4   Latonya female black    0 black female
## 5    Carrie female white    0        white
## 6       Jay   male white    0        white
```

```
resume %>%
  group_by(race, sex) %>%
  summarise(call = mean(call))
```

```
## `summarise()` has grouped output by 'race'. You can override using the `.groups` argument.
```

```
## # A tibble: 4 x 3
## # Groups:   race [2]
##   race  sex      call
##   <chr> <chr>   <dbl>
## 1 black female 0.0663
## 2 black male   0.0583
## 3 white female 0.0989
## 4 white male   0.0887
```

```
resume %>%
  group_by(firstname) %>%
  summarise(call = mean(call)) %>%
  arrange(call)
```

```
## # A tibble: 36 x 2
##    firstname   call
##    <chr>      <dbl>
##  1 Aisha     0.0222
##  2 Rasheed   0.0299
##  3 Keisha    0.0383
##  4 Tremayne  0.0435
##  5 Kareem    0.0469
##  6 Darnell   0.0476
##  7 Tyrone    0.0533
##  8 Hakim     0.0545
##  9 Tamika    0.0547
## 10 Lakisha   0.055
## # ... with 26 more rows
```

2.3 Causal Affects and the Counterfactual

```
data("social", package = "qss")
summary(social)
```

```
##      sex             yearofbirth    primary2004       messages
##  Length:305866      Min.   :1900   Min.   :0.0000   Length:305866
##  Class :character   1st Qu.:1947   1st Qu.:0.0000   Class :character
##  Mode  :character   Median :1956   Median :0.0000   Mode  :character
##                     Mean   :1956   Mean   :0.4014
##                     3rd Qu.:1965   3rd Qu.:1.0000
##                     Max.   :1986   Max.   :1.0000
##   primary2006         hhsize
##  Min.   :0.0000   Min.   :1.000
##  1st Qu.:0.0000   1st Qu.:2.000
##  Median :0.0000   Median :2.000
##  Mean   :0.3122   Mean   :2.184
##  3rd Qu.:1.0000   3rd Qu.:2.000
##  Max.   :1.0000   Max.   :8.000
```

```
turnout_by_message <-
  social %>%
  group_by(messages) %>%
  summarize(turnout = mean(primary2006))
turnout_by_message
```

```
## # A tibble: 4 x 2
##   messages   turnout
##   <chr>        <dbl>
## 1 Civic Duty   0.315
## 2 Control      0.297
## 3 Hawthorne    0.322
## 4 Neighbors    0.378
```

```
turnout_by_message %>%
  spread(messages, turnout) %>%
  mutate(diff_civic_duty = `Civic Duty` - Control,
         diff_Hawthorne = Hawthorne - Control,
         diff_Neighbors = Neighbors - Control) %>%
  select(matches("diff_"))
```

```
## # A tibble: 1 x 3
##   diff_civic_duty diff_Hawthorne diff_Neighbors
##             <dbl>          <dbl>          <dbl>
## 1          0.0179         0.0257         0.0813
```

```
social %>%
  mutate(age = 2006 - yearofbirth) %>%
  group_by(messages) %>%
  summarise(primary2004 = mean (primary2004),
            age = mean(age),
            hhisze = mean(hhsize))
```

```
## # A tibble: 4 x 4
##   messages   primary2004   age hhisze
##   <chr>            <dbl> <dbl>  <dbl>
## 1 Civic Duty       0.399  49.7   2.19
## 2 Control          0.400  49.8   2.18
## 3 Hawthorne        0.403  49.7   2.18
## 4 Neighbors        0.407  49.9   2.19
```

```
social %>%
  mutate(age = 2006 - yearofbirth) %>%
  group_by (messages) %>%
  summarise_at(vars(primary2004, age, hhsize), funs(mean))
```

```
## Warning: `funs()` was deprecated in dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
```

```
## # A tibble: 4 x 4
##   messages   primary2004   age hhsize
##   <chr>            <dbl> <dbl>  <dbl>
## 1 Civic Duty       0.399  49.7   2.19
## 2 Control          0.400  49.8   2.18
## 3 Hawthorne        0.403  49.7   2.18
## 4 Neighbors        0.407  49.9   2.19
```

2.4 Observational Studies

```
data("minwage", package = "qss")
```

```
glimpse(minwage)
```

```
## Rows: 358
## Columns: 8
## $ chain      <chr> "wendys", "wendys", "burgerking", "burgerking", "kfc", "kfc~
## $ location   <chr> "PA", "PA", "PA", "PA", "PA", "PA", "PA", "PA", "PA", "PA",~
## $ wageBefore <dbl> 5.00, 5.50, 5.00, 5.00, 5.25, 5.00, 5.00, 5.00, 5.00, 5.50,~
## $ wageAfter  <dbl> 5.25, 4.75, 4.75, 5.00, 5.00, 5.00, 4.75, 5.00, 4.50, 4.75,~
## $ fullBefore <dbl> 20.0, 6.0, 50.0, 10.0, 2.0, 2.0, 2.5, 40.0, 8.0, 10.5, 6.0,~
## $ fullAfter  <dbl> 0.0, 28.0, 15.0, 26.0, 3.0, 2.0, 1.0, 9.0, 7.0, 18.0, 5.0, ~
## $ partBefore <dbl> 20.0, 26.0, 35.0, 17.0, 8.0, 10.0, 20.0, 30.0, 27.0, 30.0, ~
## $ partAfter  <dbl> 36, 3, 18, 9, 12, 9, 25, 32, 39, 10, 20, 4, 13, 20, 15, 19,~
```

```
summary(minwage)
```

```
##     chain              location           wageBefore      wageAfter
##  Length:358         Length:358         Min.   :4.250   Min.   :4.250
##  Class :character   Class :character   1st Qu.:4.250   1st Qu.:5.050
##  Mode  :character   Mode  :character   Median :4.500   Median :5.050
##                                        Mean   :4.618   Mean   :4.994
##                                        3rd Qu.:4.987   3rd Qu.:5.050
##                                        Max.   :5.750   Max.   :6.250
##    fullBefore       fullAfter        partBefore      partAfter
##  Min.   : 0.000   Min.   : 0.000   Min.   : 0.00   Min.   : 0.00
##  1st Qu.: 2.125   1st Qu.: 2.000   1st Qu.:11.00   1st Qu.:11.00
##  Median : 6.000   Median : 6.000   Median :16.25   Median :17.00
##  Mean   : 8.475   Mean   : 8.362   Mean   :18.75   Mean   :18.69
##  3rd Qu.:12.000   3rd Qu.:12.000   3rd Qu.:25.00   3rd Qu.:25.00
##  Max.   :60.000   Max.   :40.000   Max.   :60.00   Max.   :60.00
```

```
NJ_MINWAGE <- 5.05
```

```
minwage %>%
  count(location)
```

```
##    location   n
## 1 centralNJ  45
## 2   northNJ 146
## 3        PA  67
## 4   shoreNJ  33
## 5   southNJ  67
```

```
minwage <-
  mutate(minwage, state = str_sub(location, -2L))
```

```
minwage <-
  mutate(minwage, state = if_else(location == "PA", "PA", "NJ"))
```

```
minwage %>%
  group_by(state) %>%
  summarise(prop_after = mean(wageAfter < NJ_MINWAGE), prop_Before = mean(wageBefore < NJ_MINWAGE))
```

```
## # A tibble: 2 x 3
##   state prop_after prop_Before
##   <chr>      <dbl>       <dbl>
## 1 NJ       0.00344       0.911
## 2 PA       0.955         0.940
```

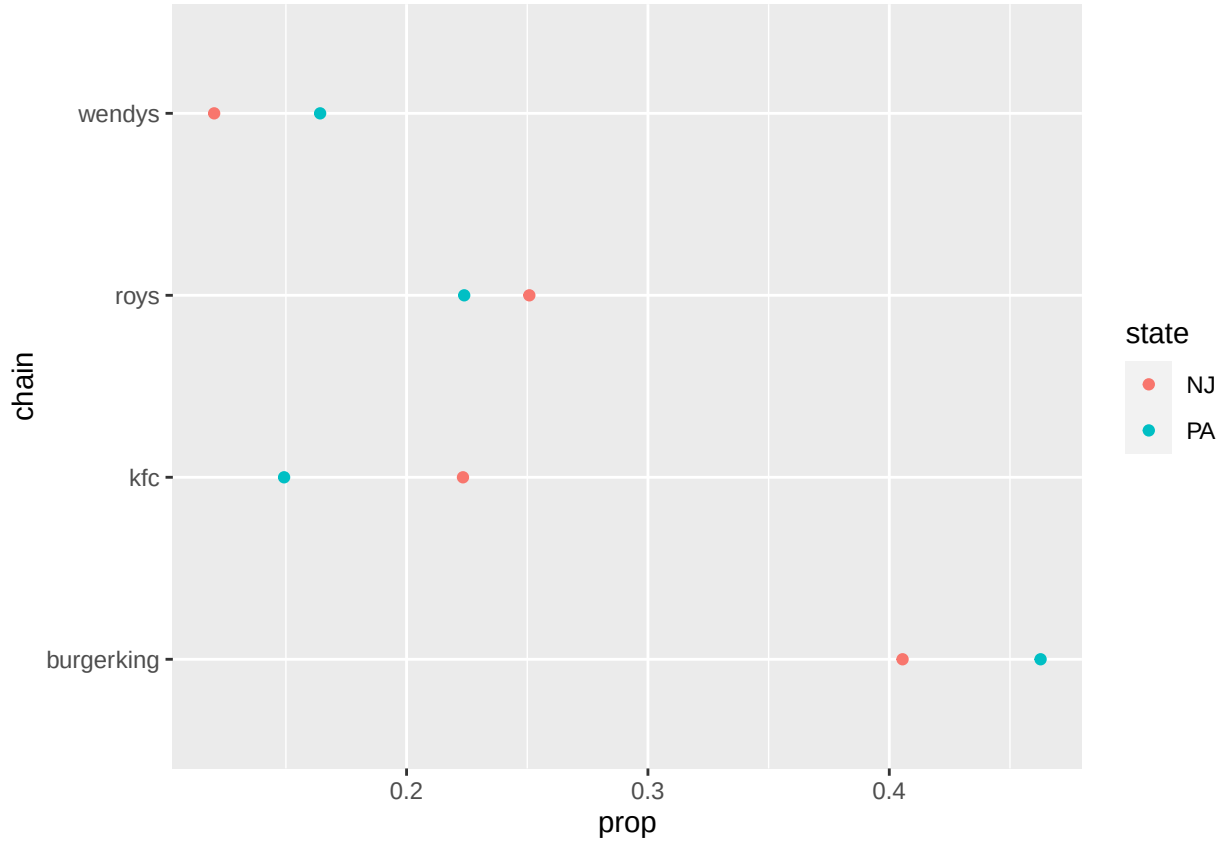
```
minwage <-
  minwage %>%
  mutate(totalAfter = fullAfter + partAfter,
         fullPropAfter = fullAfter / totalAfter)
```

```
full_prop_by_state <-
  minwage %>%
  group_by(state) %>%
  summarise(fullPropAfter = mean(fullPropAfter))
full_prop_by_state
```

```
## # A tibble: 2 x 2
##   state fullPropAfter
##   <chr>         <dbl>
## 1 NJ            0.320
## 2 PA            0.272
```

```
chains_by_state <-
  minwage %>%
  group_by(state) %>%
  count(chain) %>%
  mutate(prop = n / sum(n))
```

```
ggplot(chains_by_state, aes(x = chain, y = prop, colour = state)) +
  geom_point() +
  coord_flip()
```



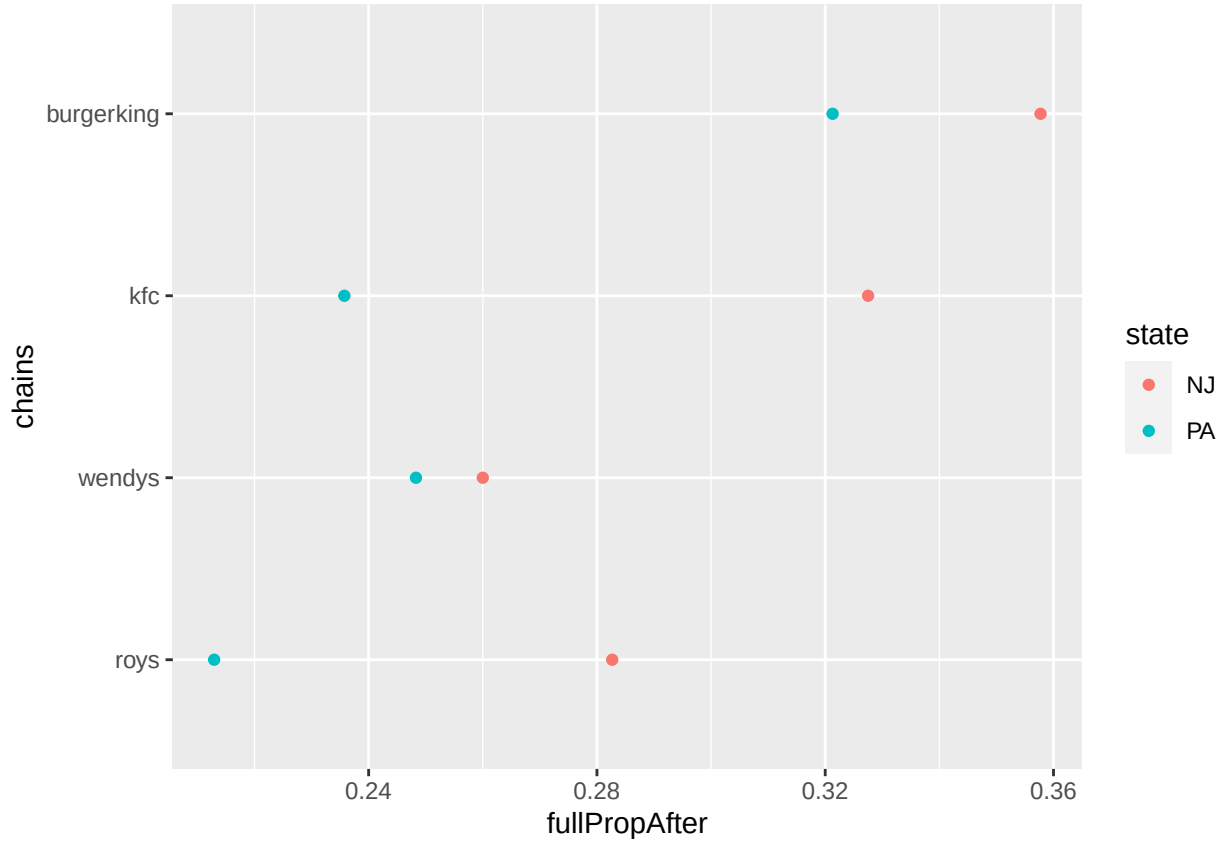
```r
full_prop_by_state_chain <-
  minwage %>%
  group_by(state, chain) %>%
  summarise(fullPropAfter = mean(fullPropAfter))
```

```
## `summarise()` has grouped output by 'state'. You can override using the `.groups` argument.
```

```r
full_prop_by_state_chain
```

```
## # A tibble: 8 x 3
## # Groups:   state [2]
##   state chain      fullPropAfter
##   <chr> <chr>              <dbl>
## 1 NJ    burgerking         0.358
## 2 NJ    kfc                0.328
## 3 NJ    roys               0.283
## 4 NJ    wendys             0.260
## 5 PA    burgerking         0.321
## 6 PA    kfc                0.236
## 7 PA    roys               0.213
## 8 PA    wendys             0.248
```

```r
ggplot(full_prop_by_state_chain,
       aes(x = forcats::fct_reorder(chain, fullPropAfter),
           y = fullPropAfter,
           colour = state)) +
geom_point() +
coord_flip() +
labs(x="chains")
```




```
full_prop_by_state %>%
  spread(state, fullPropAfter) %>%
  mutate(diff = NJ - PA)
```

```
## # A tibble: 1 x 3
##      NJ    PA   diff
##   <dbl> <dbl>  <dbl>
## 1 0.320 0.272 0.0481
```

```
minwage <-
  minwage %>%
  mutate(totalBefore = fullBefore + partBefore,
         fullPropBefore = fullBefore / totalBefore)
```

```
minwage %>%
  filter(state == "NJ") %>%
  summarise(diff = mean(fullPropAfter) - mean(fullPropBefore))
```

```
##         diff
## 1 0.02387474
```

```
  minwage %>%
  group_by(state) %>%
  summarise(diff = mean(fullPropAfter) - mean(fullPropBefore)) %>%
```

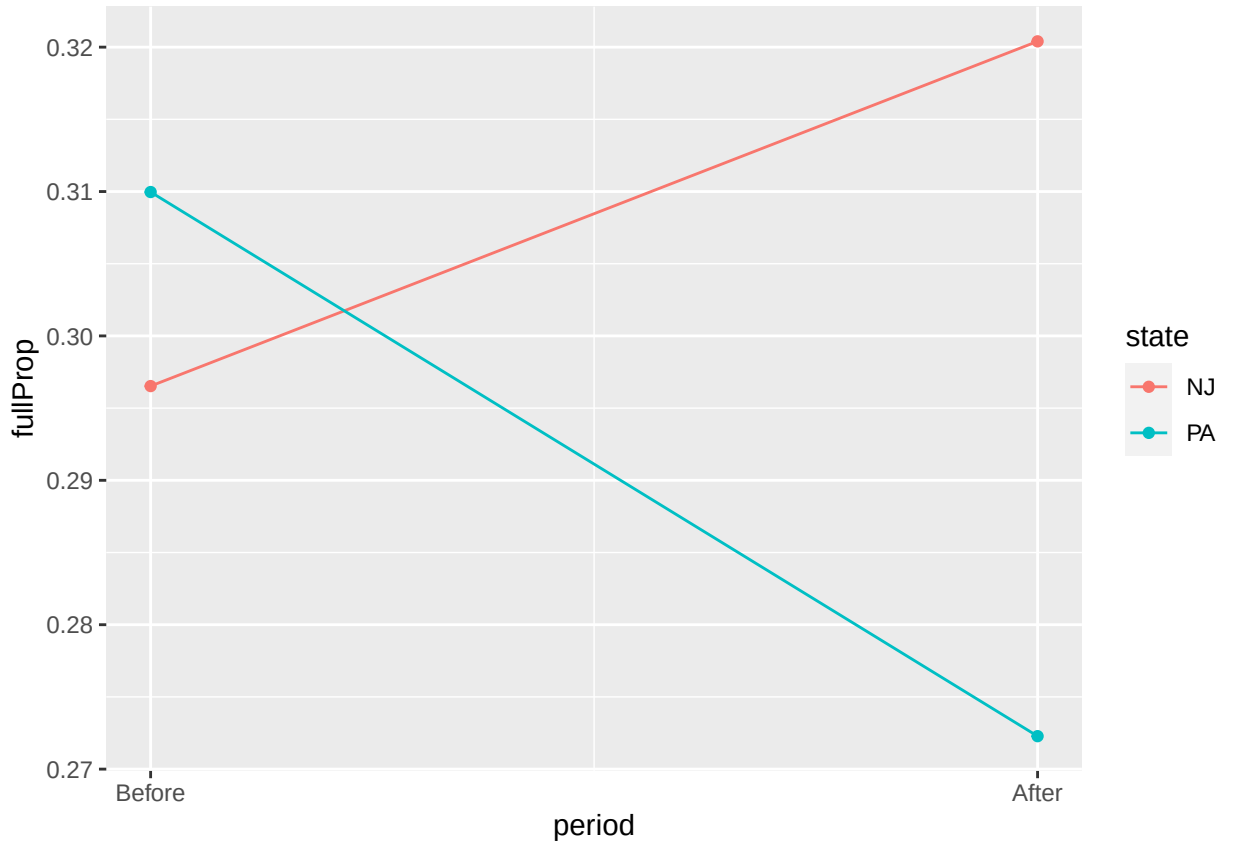
```
  spread(state, diff) %>%
  mutate(diff_in_diff = NJ - PA)
```

```
## # A tibble: 1 x 3
##       NJ      PA diff_in_diff
##    <dbl>   <dbl>        <dbl>
## 1 0.0239 -0.0377       0.0616
```

```
full_prop_by_state <-
  minwage %>%
  group_by(state) %>%
  summarise_at(vars(fullPropAfter, fullPropBefore), mean) %>%
  gather(period, fullProp, -state) %>%
  mutate(period = recode(period, fullPropAfter = 1, fullPropBefore = 0))
full_prop_by_state
```

```
## # A tibble: 4 x 3
##   state period fullProp
##   <chr>  <dbl>    <dbl>
## 1 NJ         1    0.320
## 2 PA         1    0.272
## 3 NJ         0    0.297
## 4 PA         0    0.310
```

```
ggplot(full_prop_by_state, aes (x = period, y = fullProp, colour = state)) +
  geom_point() +
  geom_line() +
  scale_x_continuous(breaks = c(0,1), labels = c("Before", "After"))
```



##2.5 Descriptive Statistics for a Single Variable

```
minwage %>%
  filter(state == "NJ") %>%
  select(wageBefore, wageAfter) %>%
  summary()
```

```
##    wageBefore     wageAfter
##  Min.   :4.25   Min.   :5.000
##  1st Qu.:4.25   1st Qu.:5.050
##  Median :4.50   Median :5.050
##  Mean   :4.61   Mean   :5.081
##  3rd Qu.:4.87   3rd Qu.:5.050
##  Max.   :5.75   Max.   :5.750
```

```
minwage %>%
  group_by(state) %>%
  summarise(wageAfter = IQR(wageAfter),
            wageBefore = IQR(wageBefore))
```

```
## # A tibble: 2 x 3
##   state wageAfter wageBefore
##   <chr>     <dbl>      <dbl>
## 1 NJ        0           0.62
## 2 PA        0.575       0.75
```

```
minwage %>%
  group_by(state) %>%
  summarise(wageAfter_sd = sd(wageAfter),
            wageAfter_var = var(wageAfter),
            wageBefore_sd = sd(wageBefore),
            wageBefore_var = var(wageBefore))
```

```
## # A tibble: 2 x 5
##   state wageAfter_sd wageAfter_var wageBefore_sd wageBefore_var
##   <chr>        <dbl>         <dbl>         <dbl>          <dbl>
## 1 NJ           0.106        0.0112         0.343          0.118
## 2 PA           0.359        0.129          0.358          0.128
```

```
minwage %>%
  group_by(state) %>%
  summarise_at(vars(wageAfter, wageBefore), funs(sd, var))
```

```
## # A tibble: 2 x 5
##   state wageAfter_sd wageBefore_sd wageAfter_var wageBefore_var
##   <chr>        <dbl>         <dbl>         <dbl>          <dbl>
## 1 NJ           0.106         0.343        0.0112          0.118
## 2 PA           0.359         0.358        0.129           0.128
```