BUSN 5000 Part I Project

Getting started

Peng Zhengdong

Section 1: Missing Data

Suppose we have data on a sample of working people. W_i^* is the actual wage of person i, W_i is the wage the person reports in the survey (including the possibility they did not respond) and $R_i = 1$ if they reported a value for their wage and $R_i = 0$ if they did not. Assume $W_i = W_i^*$ if $R_i = 1$ and $W_i = NA$ if $R_i = 0$.

- **Section 1: Question 1** Using the Law of Iterated Expectations, write an equation that expresses $E[W^*]$ in terms of $E[W^*|R_i = 1]$, $E[W^*|R_i = 0]$, $Pr(R_i = 1)$ and $Pr(R_i = 0)$.

Answer

$E[W^*] = E[W^*|R_i = 1]Pr(R_i = 1) + E[W^*|R_i = 0]Pr(R_i = 0)$

- **Section 1 Question 2** In the equation you just wrote, which of $E[W^*|R_i = 1]$, $E[W^*|R_i = 0]$, $Pr(R_i = 1)$ and $Pr(R_i = 0)$ are *observable* and which are *unobservable*?

Answer

- $E[W^*|R_i = 1]$ is *observable*
- $E[W^*|R_i = 0]$ is *unobservable*
- $Pr(R_i = 1)$ is *observable*
- $Pr(R_i = 0)$ is *observable*

- **Section 1 Question 3** Under what *one circumstance* is it possible to learn $E[W^*]$ from observed data *without making any further assumptions*?

Answer

$E[W^*] = E[W^*|R_i = 1]$ which can only true if $Pr(R_i = 1)$ =1 or $Pr(R_i = 0)$ =0 (means no missing data)

- **Section 1 Question 4** Suppose you know that $Pr(R_i = 1) = 0.5$ and $E[W|R_i = 1] = 20$. If you are willing to assume that $E[W^*|R_i = 0]$ is between 10 and 30, what is the possible range of values for $E[W^*]$? Show your work.

Answer

$E[W^*]$ = $E[W|R_i = 1]$ x $Pr(R_i = 1)$ + $E[W|R_i = 0]$ x $Pr(R_i = 0)$.

The probability $R_i = 1$ and $R_i = 0$ are both 0.5.

So, the $E[W^*]$ is maxmium when $E[W^*|R_i = 0]$ = 30, and is mimium when $E[W^*|R_i = 0]$ = 10.

When $E[W^*|R_i = 0]$ = 10, $E[W^*]$ = 20x0.5+10x0.5 = 15

When $E[W^*|R_i = 0]$ = 30, $E[W^*]$ = 20x0.5+30x0.5 = 25

- The minimum value for $E[W^*]$ is 15
- The maximum value for $E[W^*]$ is 25

- **Section 1 Question 5** What must you assume for an estimate of $E[W|R_i = 1]$ to be an unbiased estimate for $E[W^*]$?

Answer

For $E[W|R_i = 1]$ to be an unbiased estimate for $E[W^*]$, $E[W^*|R_i = 0]$ must equal to $E[W|R_i = 1]$

Section 2: The role of non-startups in job growth

Draw on the code from `Census_Blog_Replication.Rmd` to write a reproducible analysis that will generate a plot of the number of new jobs created by firms that *are not* startups (also called *continuing firms*) as a percentage of total employment.

Answer

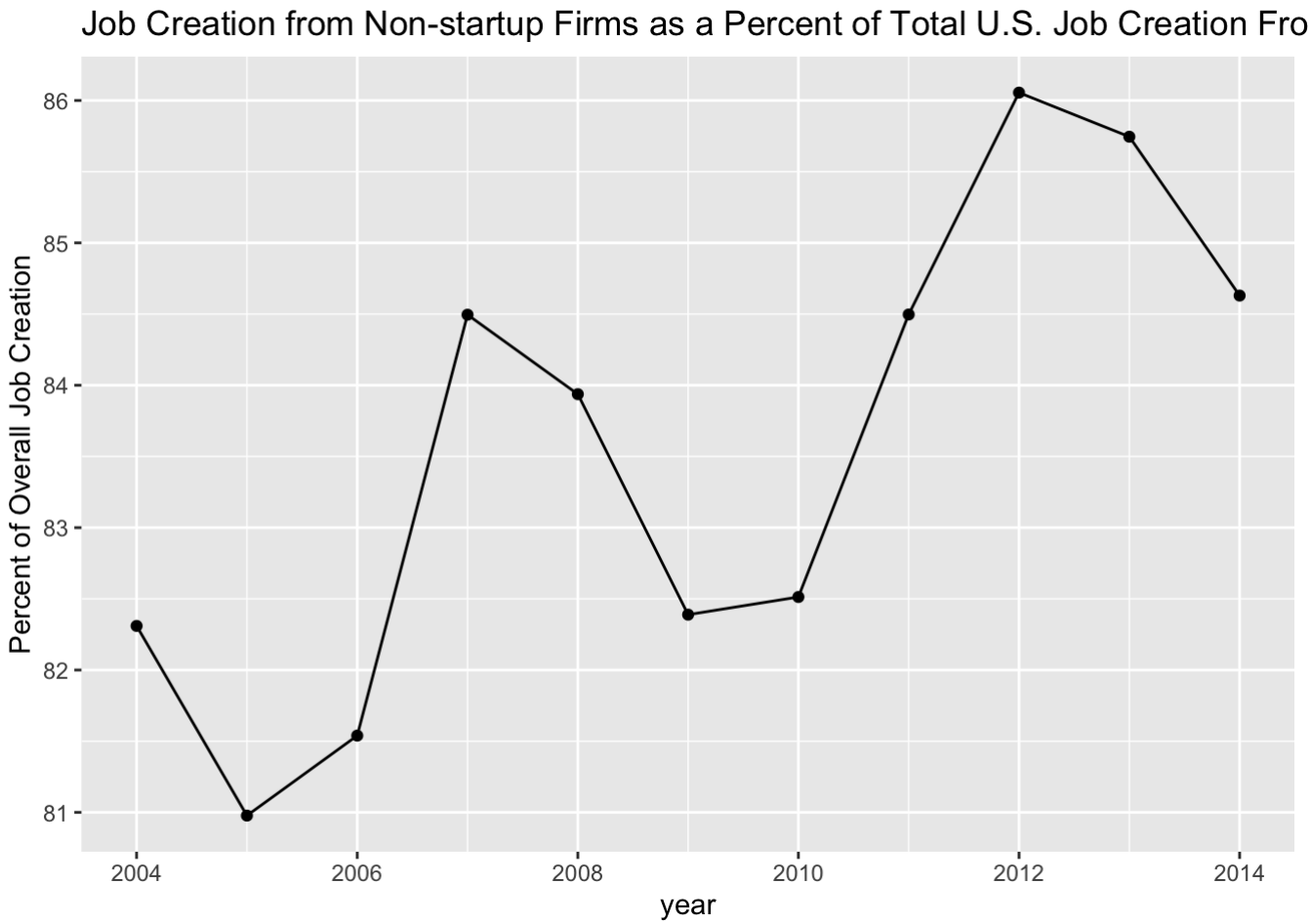
```
# Make sure to set the above to read eval = TRUE before you try to knit
ewfile <- "https://www2.census.gov/ces/bds/firm/bds_f_all_release.csv"
fafile <- "https://www2.census.gov/ces/bds/firm/bds_f_age_release.csv"
ewdata <- read_csv(ewfile)
```

```
## Rows: 38 Columns: 25
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## dbl (25): year2, firms, estabs, emp, denom, estabs_entry, estabs_entry_rate,...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
fadata <- read_csv(fafile)
```

```
## Rows: 361 Columns: 28
## ── Column specification ────────────────────────────────────────────────────
## Delimiter: ","
## chr  (1): fage4
## dbl (26): year2, Firms, Estabs, Emp, Denom, Estabs_Entry, Estabs_Entry_Rate,...
## lgl  (1): sic1
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
total_data <- ewdata %>%
  select(year2,job_creation, emp) %>%
  rename(year = year2,
         jc_total = job_creation,
         emp_total = emp)

startup_data <- fadata %>%
  filter(fage4 == "a) 0") %>%                    # only keep rows for startups
#
  rename(jc_startup = Job_Creation,              # Rename variables
        year = year2) %>%
  select(year,jc_startup)                        # keep only the year and job creatio
n variables


analysis_data <- inner_join(total_data, startup_data, by = "year") %>%
  mutate(jc_nonstartup = jc_total - jc_startup,
         jc_share = 100*jc_nonstartup / jc_total,
         emp_share = 100*jc_nonstartup / emp_total) %>%
  filter(year > 2003)


non_startup_plot <- ggplot(data = analysis_data, aes(x = year, y = jc_share)) +
  geom_line() +
  geom_point() +
  ylab("Percent of Overall Job Creation") +
  ggtitle("Job Creation from Non-startup Firms as a Percent of Total U.S. Job Creatio
n From 2004 to 2014")

non_startup_plot
```



Section 3: Updating a reproducible analysis

In Fall 2020, the Census Bureau released a redesigned version of the BDS. Some of the features of the redesign are described here (https://www2.census.gov/programs-surveys/bds/updates/bds2018-release-note.pdf). We want to redo the analysis from Lawrence's post using the redesigned and updated data.

Link to the Data

Here are links to the redesigned economy-wide and firm age data.

```
# URLs to the redesigned data:

## Economy-wide data
ewfile <- "https://www2.census.gov/programs-surveys/bds/tables/time-series/bds2019.csv"

## Firm-age data
fafile <- "https://www2.census.gov/programs-surveys/bds/tables/time-series/bds2019_fac.csv"


ewdata <- read_csv(ewfile)
```

```
## Rows: 42 Columns: 25
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## dbl (25): year, firms, estabs, emp, denom, estabs_entry, estabs_entry_rate, ...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
fadata <- read_csv(fafile, na = c("(X)","(S)", "(D)"))
```

```
## Rows: 210 Columns: 26
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr  (1): fagecoarse
## dbl (25): year, firms, estabs, emp, denom, estabs_entry, estabs_entry_rate, ...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(ewdata)
```

```
## # A tibble: 6 × 25
##    year   firms  estabs      emp   denom estab…¹ estab…² estab…³ estab…⁴ job_c…⁵
##   <dbl>   <dbl>   <dbl>    <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1  1978 3557994 4310626 69410001  6.68e7  654226    15.5  487202    11.5  1.47e7
## 2  1979 3692077 4472108 73848234  7.17e7  630253    14.3  477537    10.9  1.41e7
## 3  1980 3739809 4533251 74109267  7.40e7  592484    13.2  532203    11.8  1.22e7
## 4  1981 3770852 4615479 75728652  7.49e7  606853    13.3  522047    11.4  1.29e7
## 5  1982 3720273 4598769 74922226  7.53e7  572030    12.4  589368    12.8  1.21e7
## 6  1983 3829596 4713538 74178554  7.45e7  621854    13.4  507458    10.9  1.26e7
## # … with 15 more variables: job_creation_births <dbl>,
## #   job_creation_continuers <dbl>, job_creation_rate_births <dbl>,
## #   job_creation_rate <dbl>, job_destruction <dbl>,
## #   job_destruction_deaths <dbl>, job_destruction_continuers <dbl>,
## #   job_destruction_rate_deaths <dbl>, job_destruction_rate <dbl>,
## #   net_job_creation <dbl>, net_job_creation_rate <dbl>,
## #   reallocation_rate <dbl>, firmdeath_firms <dbl>, firmdeath_estabs <dbl>, …
```

```
head(fadata)
```

```
## # A tibble: 6 × 26
##    year fageco…¹   firms  estabs     emp   denom estab…² estab…³ estab…⁴ estab…⁵
##   <dbl> <chr>      <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1  1978 a) 0      485415  493592  2.58e6  1.29e6  493592  200.        NA     NA
## 2  1978 b) 1 to…      NA      NA  2.40e6  2.39e6    4831    1.13  124648    29.2
## 3  1978 c) 6 to…      NA      NA NA      NA           NA   NA         NA     NA
## 4  1978 d) 11+        NA      NA NA      NA           NA   NA         NA     NA
## 5  1978 e) Left… 2717082 3449662  6.44e7  6.31e7  155803    4.38  362549    10.2
## 6  1979 a) 0      473488  478317  2.46e6  1.23e6  478317  200.        NA     NA
## # … with 16 more variables: job_creation <dbl>, job_creation_births <dbl>,
## #   job_creation_continuers <dbl>, job_creation_rate_births <dbl>,
## #   job_creation_rate <dbl>, job_destruction <dbl>,
## #   job_destruction_deaths <dbl>, job_destruction_continuers <dbl>,
## #   job_destruction_rate_deaths <dbl>, job_destruction_rate <dbl>,
## #   net_job_creation <dbl>, net_job_creation_rate <dbl>,
## #   reallocation_rate <dbl>, firmdeath_firms <dbl>, firmdeath_estabs <dbl>, …
```

Note that the redesign involved a change to some variable names. Note also that the `bds2019_fac.csv` uses the character strings (X), (S), and (D) as *data quality flags* that indicate values that are either missing or have been suppressed. These changes are built into the code chunk

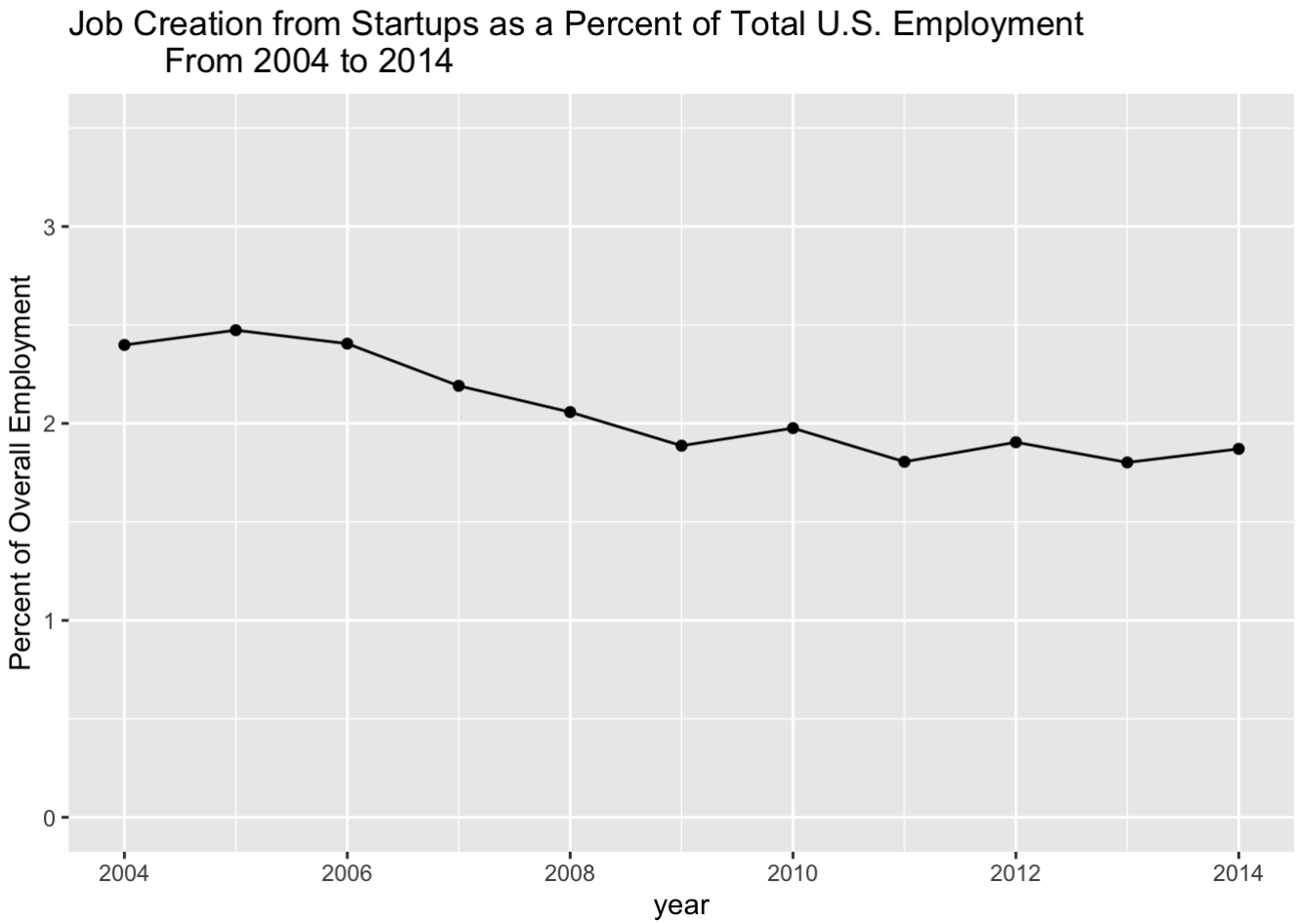
Documenting the new data

- **Section 3: Question 1** Consult the codebook for the new BDS (https://www.census.gov/content/dam/Census/programs-surveys/business-dynamics-statistics/codebook-glossary.pdf). Describe what each of the the *data quality flags* means
- **Answer**
    - (X): A structurally missing flag will appear as (X), when cells are structurally zero or structurally missing.
    - (D): A Disclosure suppression will appear as (D) wjem a cell has too few firms.
    - (S): A Data quality suppression will appear as (s) When a cell is determined to be unreliable due to its time series characteristics.

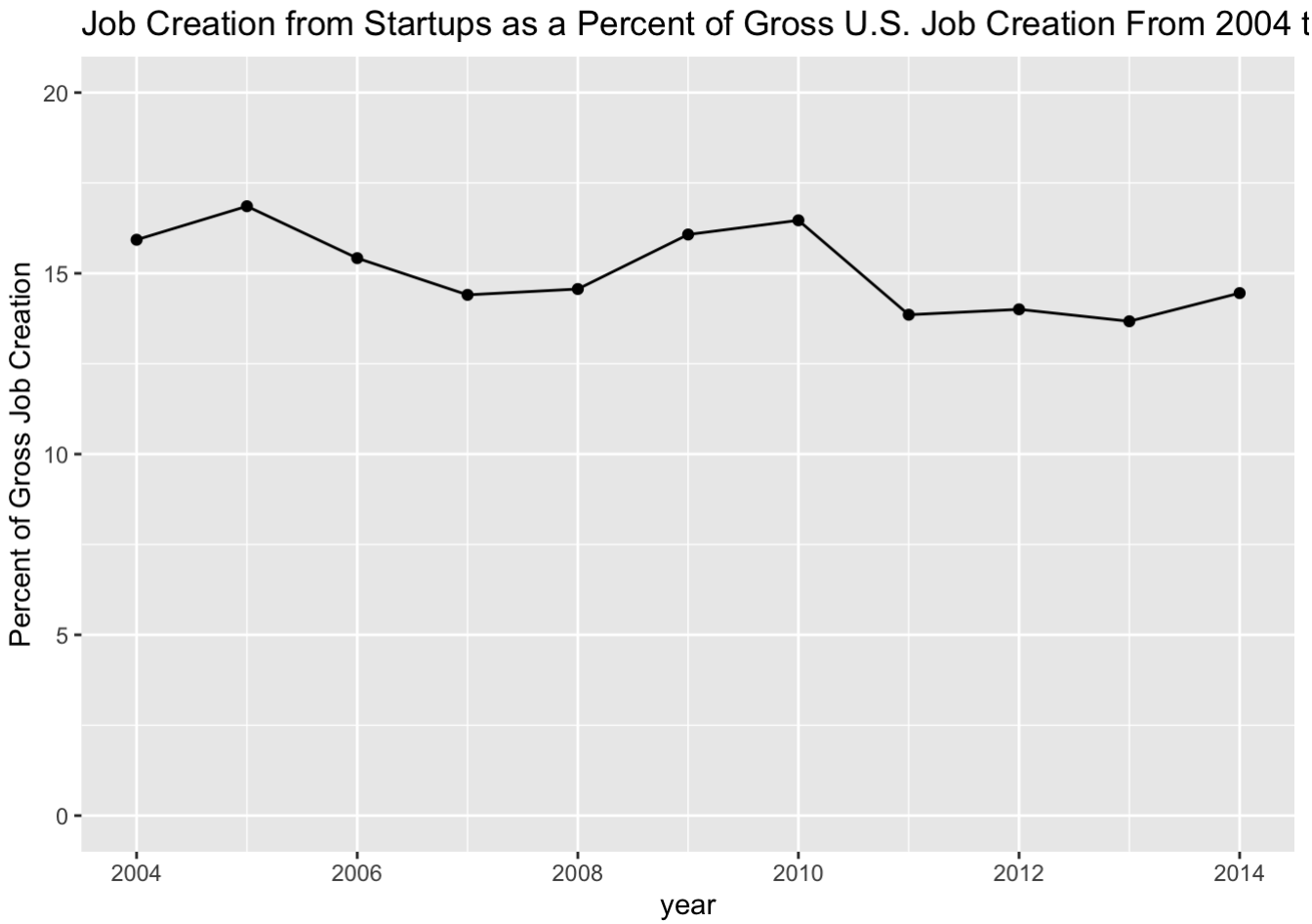
Updating the Census Blog Post

- Now attempt to reproduce Lawrence's plots using the redesigned BDS data. Specifically, draw on `Census_Blog_Replication.Rmd` to help edit the code chunks below so they will generate Lawrence's plots, but on the updated data.

```
# Create a new tibble data frame that keeps only the year, job_creation and
# employment (emp) variables from the economy-wide data.
total_data <- ewdata %>%
  select(year,job_creation, emp) %>%
  rename(year = year,
         jc_total = job_creation,
         emp_total = emp)

# Create data frame that keeps only the year, job_creation and employment (emp)
# variables on observations for startups.
startup_data <- fadata %>%
  filter(fagecoarse == "a) 0") %>%  # only keep rows for startups
  rename(jc_startup = job_creation,              # Rename variables
        year = year) %>%  #format. Details later
  select(year,jc_startup)   # keep only the year and job creation variables

analysis_data <- inner_join(total_data, startup_data, by = "year") %>%  # join the da
ta by year
  mutate(emp_share = 100* jc_startup / emp_total , # construct the analysis variables
         jc_share = 100 *jc_startup / jc_total)

# Keep only observations between 2004 and 2014
plot_data <- analysis_data %>%
  filter(year >= 2004 & year <= 2014)


emp_share_plot <- ggplot(data = plot_data ,
                         mapping = aes(x=year,y=emp_share)) +
  geom_line()+
  geom_point() +
  ylab("Percent of Overall Employment") +
  ylim(0,3.5) +
  ggtitle("Job Creation from Startups as a Percent of Total U.S. Employment
          From 2004 to 2014")

emp_share_plot   ## This statement displays the plot object we just created
```



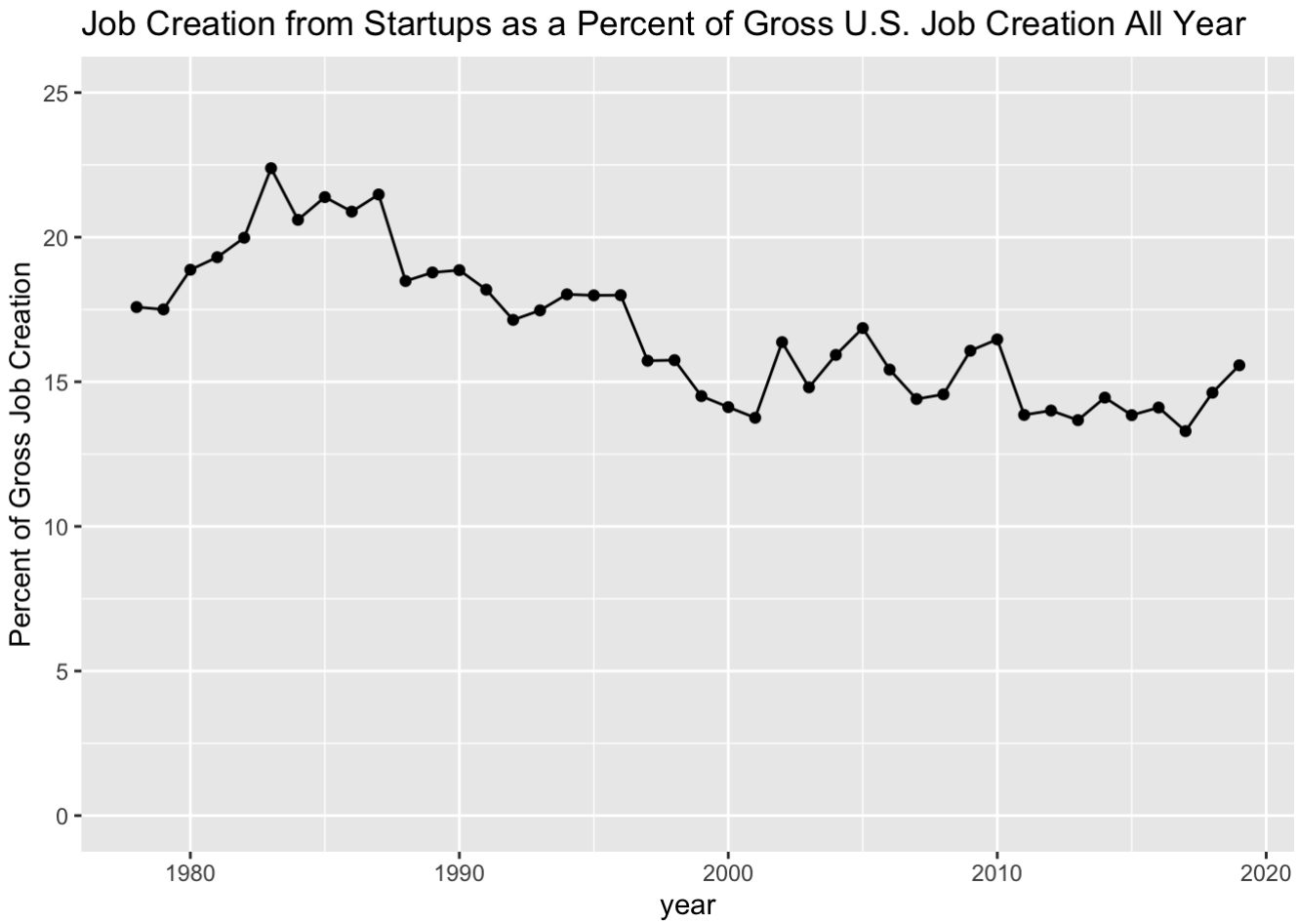

```
jc_share_plot <- ggplot(data = plot_data ,
                        mapping = aes(x=year,y=jc_share)) +
  geom_line()+
  geom_point() +
  ylab("Percent of Gross Job Creation") +
  ylim(0,20) +
  ggtitle("Job Creation from Startups as a Percent of Gross U.S. Job Creation From 20
04 to 2014")

jc_share_plot  ## This statement displays the plot object we just created
```



- Make a plot of job creation from startups as a percent of gross job creation for **all years** in the redesigned data. Hint: you just need to take the code for `jc_share_plot` and apply it to `analysis_data` instead of `plot_data`. Insert such code into the block below

Answer

```
jc_share_plot_allyears <- ggplot(data = analysis_data,
                                 mapping = aes(x=year,y=jc_share)) +
  geom_line()+
  geom_point() +
  ylab("Percent of Gross Job Creation") +
  ggtitle("Job Creation from Startups as a Percent of Gross U.S. Job Creation All Yea
r")+
  ylim(0,25)

jc_share_plot_allyears
```



Section 4: Bayes' Rule

Spam filter

SpamAssassin works by having users train their email program to recognize spam. The program studies emails that have been marked as spam by the user.

Suppose that based on the user-provided data, the program finds the following three patterns:

- The word "Free" appears in 30 percent of emails marked as spam
- The word "Free" appears in 2 percent of emails marked as not spam
- 70 percent of all messages are marked as spam

Section 4: Question 1

Assume that the email program uses Bayes' Rule to determine whether a given message is spam. What is the probability of being spam that SpamAssassin assigns to an email with the word "Free"

Translation:

- $P(Free \mid Spam)$ = 30%
- $P(Free \mid NotSpam)$ = 2%
- $P(Spam)$ = 70%
- $P(NotSpam)$ = 1- 70% = 30%
- Baye's theorem: $P(A \mid B) = P(B \mid A)^{*}P(A) \,/\, P(B)$
- $P(Free)$ = $(P(Spam)P(Free \mid Spam))$ + $(P(NotSpam)^{*}P(Free \mid NotSpam))$ =70% x 30% + 30% x 2% = 21.6%

Find $P(Spam \mid Free)$

- $P(Spam \mid Free)$ = $P(Free \mid Spam)$*$P(Spam)$ / $P(Free)$ = 30% x 70% / 21.6% = 97.22%

Section 4: Question 2

Now assume the program also knows that

- The word “Opportunity” appears in 20 percent of emails marked as spam
- The word “Opportunity” appears in 5 percent of emails not marked as spam

Assume that whether the word “Opportunity” appears is independent of whether the word “Free” appears. What is the probability of being spam that SpamAssassin assigns to an email containing both the word “Free” that does not contain the word “Opportunity”?

[HINT: Build on your answer to the previous question]

Translation:

*$P(Opp \mid Spam)$ = 20%

*$P(Opp \mid NotSpam)$ = 5%

*$P(NotOpp \mid Spam)$ = 1- 20% = 80%

*$P(Opp \mid NotSpam)$ = 1 - 5% = 95%

*$P(Opp)$ = $P(Spam)$x$P(Opp \mid Spam)$ + $P(NotSpam)$x $P(Opp \mid NotSpam)$ = 70%x20% + 30%x5% = 15.5%

- $P(NotOpp)$ = 1-15.5% = 84.5%

Find

*$P(Spam \mid NotOpp, Free)$ = $P(Spam)$ x $P(NotOpp \mid Spam)$ x $P(Free \mid Spam)$ / ($P(NotOpp)P(Free)$)

- = 0.7x0.8x0.3 / 0.845x0.216 = 92.04%