Data Scientist Job Postings in New York City: Educational and Technical/Analytical Skills, and How they Relate to Estimated Salaries

Nikita Stempniewicz

October 24, 2017

Introduction

In general, a data scientists' primary function is extracting insights from data, and communicating those insights in a clear and efficient manner. This requires a diverse skill set based in math, statistics, and computer, and information sciences.(Wikipedia, the free encyclopedia 2017) There is no typical technical or educational skill set required of data scientist, and different employers value different backgrounds. Some companies only require a bachelor's degree while others require a PhD for their data scientist. Most require programming and statistics skills, but often differ in the programming language or statistical software, e.g., SQL, R, STATA, Python, etc. Some data scientist positions are more focused on data visualization and communication, with an emphasis on software such as tableau, and others on exploratory analysis and more of an emphasis on statistical methods such as machine learning.(Strauss 2017, Ramel (2016), Pierson (2017). Burning Glass Technologies (2017))

IBM and Burning Glass Technologies recently released a report, The Quant Crunch: How the Demand for Data Science Skills is Disrupting the Job Market, where it projected the demand for data and analytic talent to increase broadly by 364,000 openings, to 2,720,000 by 2020, with the fastest growth among data scientists and advanced analysts (27% increase by 2020). The same report found the average salary of data scientist to be $94,576, and in general, salaries for all data and analytic talent changes significantly depending on the educational and analytical skills desired of the position. Finally, the same report listed the top four metro areas in the United States for data scientist and advanced analytics job postings as New York City (NYC), San Francisco, Washington D.C., and Boston, respectively. (Burning Glass Technologies 2017)

Current job listings for data scientist positions can be found on online job boards, e.g., glassdoor, and queried by location. These often include information on desired education and technical/analytic skills, in addition to an estimated salary. Glassdoor calculates estimated salary using machine learning algorithms based on data from millions of employees, and focuses on key factors such as job title, employer, and job location. (Glassdoor 2017)

By web scraping job postings for data scientist in NYC from Glassdoor, we were able to obtain over 700 job descriptions and estimated salaries. Using this data, we described estimated salaries, desired education, and technical/analytical skills for data scientist positions in the NYC metro area. Beyond our descriptive analyses we looked at the relationship of desired education and technical/analytical skills with estimated salary.

We found in New York City, there was no typical education or technical/analytical background desired of data scientists, and differences in salaries were associated with those different background.

Data Source & Methods

To minimize the variation in salaries due to geographic differences, the location when searching for data scientist positions was restricted to New York City, which includes the city itself and surrounding metro areas, and was ranked highest for data scientist and advanced analytics job postings in the Quant Crunch report. We acknowledge a limitation of the study is that the results from this analysis are most relevant and generalizable to data scientist positions in NYC and other similar cities, and feel the strength of removing any geographic differences in salary justified our decision.

Scraping Glassdoor

All the data for this analysis was scraped from Glassdoor.com in R, on October 8th 2017, using the rvest (Wickham 2016) package, and selectorgadget to identify relevant html nodes. Glassdoor includes 33 pages of results for a given search with 30 job listings per page, which totals to 990 job listings per search.

To get our data, first we built the URLs with a loop for the 33 different summary pages. Next, we scraped all the html data from the websites and extracted relevant data for the 30 posts on each summary page using the appropriate html nodes found using selectorgadget. For each post we extracted (when available) job id, job title, salary, and job location, from the html code from each summary pages. Next, using the job id from the search results in the previous step, and another loop, we built the URLs for the individual job posts on glassdoor. With the URL and similar web scraping methods previously described, we extracted the raw text job descriptions from the 990 websites for the individual job postings. (see supplemental code section 1) The code from this section in the supplemental was based on code originally written by Stephen Cristiano and he authors are grateful for the contributions.

Cleaning Text Data

Finally, using text mining and analysis packages in R, i.e., Stringr (Wickham 2017), and tidytext (Silge and Robinson 2016), we used regular expressions to create structured fields for different educational (Bachelors, Masters, and Doctorate), and technical (Statistics, Research, Machine Learning, Big Data, Optimization, Data Management, Software Development, Natural Language Processing) and analytical skills (Python, SQL, R, Hadoop, Java, C, SAS, Excel, Tableau, Matlab, Linux, SPSS, Oracle, Ruby, STATA) from the raw job descriptions. The patterns used to identify and define the different skills can be found in supplemental code section 2.

Statitical Methods

Using simple statistical methods, e.g., t-tests, analysis of variance, and linear regression, we described differences in estimated salaries, desired education, and technical/analytical skills among data scientist positions in NYC, and how these attributes relate to one another.

Results

From the 990 data scientist positions in the New York City area from the original search on glassdoor, 242 were excluded from the analyses because there was no estimated salary provided, and 1 was excluded for no job description. For this analysis we used the 747 data scientist positions in the New York City area with a job description and estimated salary.

Figure 1 shows the distribution of estimated salary for the 747 data scientist positions in the New York City area. Overall, the average estimated salary for data scientist in the New York City area was 179.9 thousand dollars and the standard deviation 57.3, with the overall distribution slightly skewed to the right.

Education

It is not uncommon for employers to have required and preferred qualifications which often include multiple possibilities for educational requirements, e.g., requiring a bachelors at a minimum, but specifying a preference for candidates with a masters or PhD. When more than one educational requirement is mentioned we classify education at the highest level, e.g., posts that included BS and MS are considered MS.

Overall, we were not able to get education for 13.3% of job posts, and 23.4%, 34.1%, 29.2%, were classified as bachelors, masters, and PhD degrees respectively. Data science positions that desired a PhD education

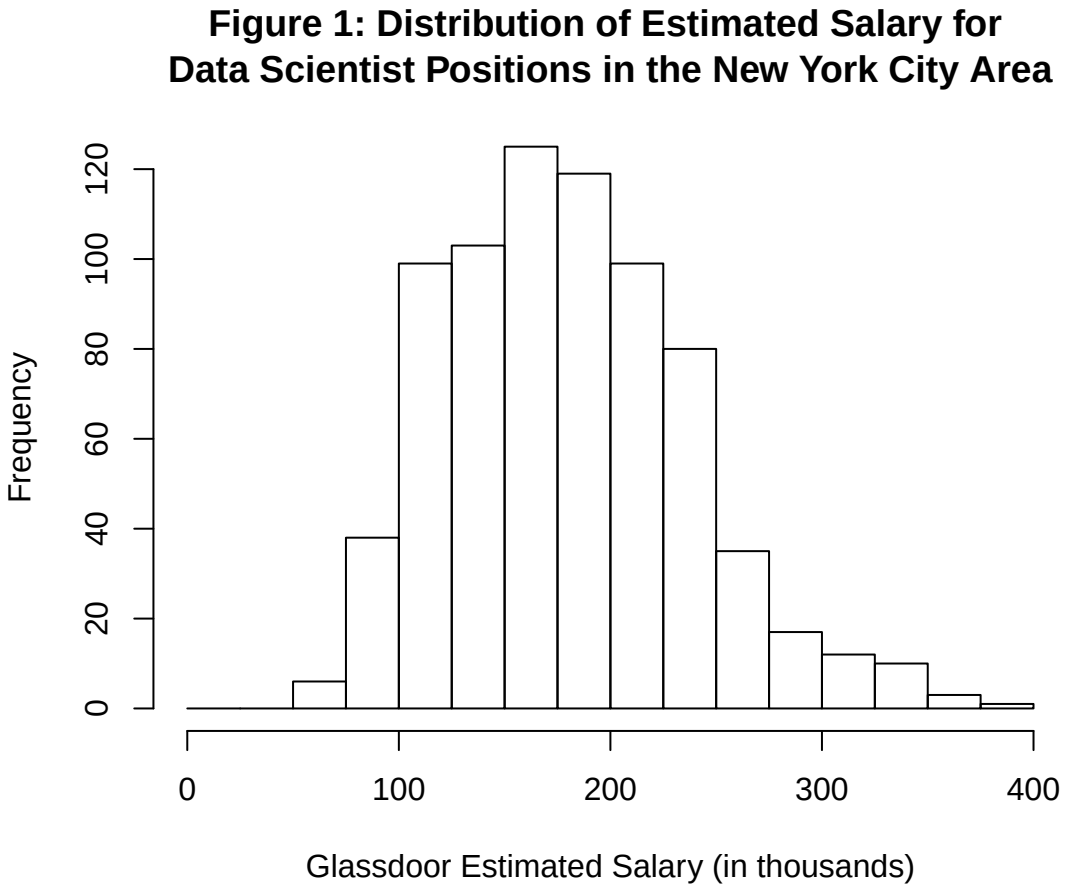

Figure 1: Distribution of estimated salaries for 747 data scientist positions, slightly skewed to the right

Table 1: Education Requirements (Degree) and Estimated Salaries (in thousand), Average (SD)

Degree	Jobs	Salaries
No Education	99 (13.3%)	177.7 (59.7)
Bachelors (BS)	175 (23.4%)	165.2 (50.8)
Masters (MS)	255 (34.1%)	179.9 (56.9)
PhD	218 (29.1%)	192.8 (59)

had an average salary of $192,800, compared to $179,900 for masters, and $165,200 when only a bachelors degree was mentioned. For jobs where no information on education was ascertained, the average salary was similar to the overall average. Figure 2 shows the differences in the medians, 25th and 75th percentiles for the different categories of education.

We used a t-test to compare the salaries for the 648 positions where education was ascertained from the job description to the 99 positions where no such information was found, and found no difference in the average salaries between the two groups (p-value=0.69). We used Analysis of Variance (ANOVA) methods to compare the average salaries between education categories, i.e., Bachelors vs. Masters vs. P.h.D, which provided enough evidence to suggest the differences in salaries are statistically significant between at least 2 of the 3 groups (p-value <.001). Statistically significant differences were confirmed and quantified using linear regression, where compared to jobs where bachelors degree is the only educational requirement mentioned, jobs that mentioned masters degrees had an estimated salary 14.7 (SE: 2.7) thousand higher, and jobs that mentioned a PhD had an estimated salary 27.7 (SE: 4.9) thousand higher, both coefficients with p-values < .01.

Job Skills

Overall, 97.6% of the 747 data scientist job postings had at least 1 of the 15 skills that were investigated. Figure 3 shows overall, the most common skills are statistics (63.3%), python (57.2%), SQL (56.4%), research (52.6%), machine learning (44.4%), and R (43.8%). Some less common skills include natural language processing (NLP, 5.9%), Ruby (3.2%), UX (2.1%), and STATA (1.3%). (figure 3)

Job Skills and Education

Table 2 summarizes the top 6 skills for data scientists by education requirements, results from ANOVA tests suggested a statistically significant difference in the proportion of job postings listing the individual skills by education for all 6 skills. In general, job postings that mention higher levels of education, i.e., Masters, or PhD, also are more likely to mention statistics, python, research, machine learning, and R, and less likely to mention SQL.

Table 2: Differnces in Top 6 Skills by Education

Skills	Bachelors	Masters	PhD	PValue
Statistics	54.90	61.20	79.40	<.01
Python	49.10	56.50	68.30	<.01
SQL	70.30	60.00	42.20	<.01
Research	53.10	47.80	74.30	<.01
Machine Learning	22.30	43.10	69.30	<.01
R	36.60	42.70	56.00	<.01

Figure 2: Differences in Estimated Salaries by Education

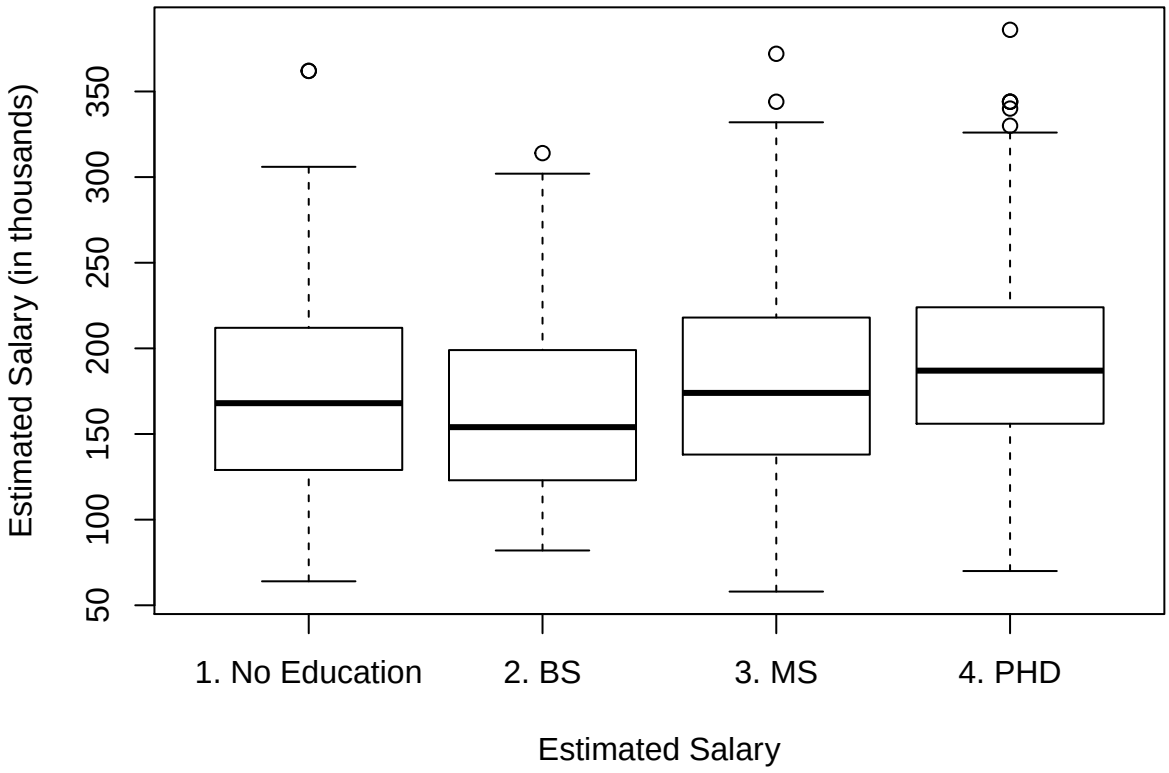

Figure 2: Distribution of estimated salaries by Education, incduing the median (thick black line), 25th percentile (bottom of box), and 75th percentile (top of box) of estimated salary. Overall, the median, 25th, and 75th percentiles of estimated salary increase with education degrees

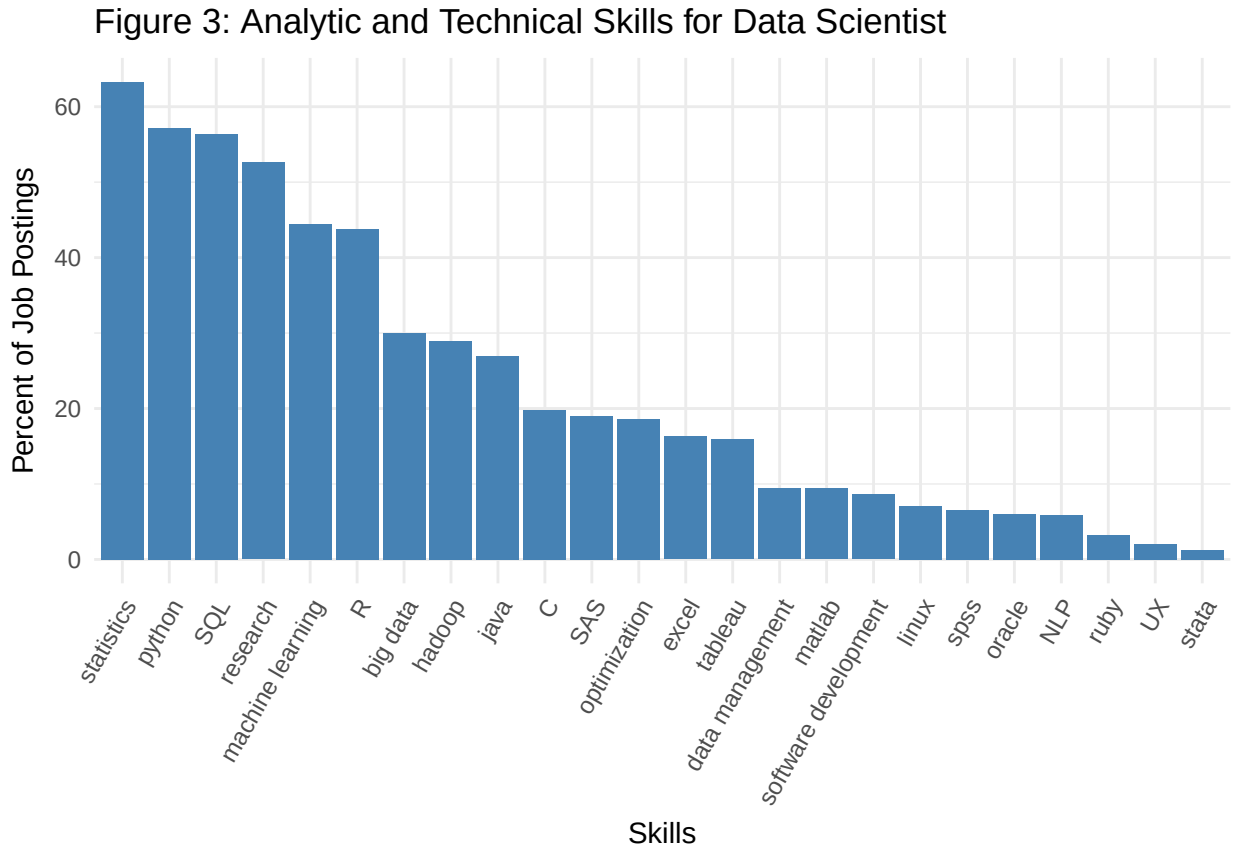

Figure 3: Percent of data scientist positions in New York City listing the specific skill, skills are ranked from highest to lowest demand

Table 3: Results From Linear Regression Models- Estimated Changes in Salary by Skill

Skills	Estimated.Salary.Increase	Standard.Error	p.value
machine learning	27.02	4.63	0.00
C	25.91	5.49	0.00
python	18.44	4.47	0.00
java	16.75	4.93	0.00
statistics	16.73	4.70	0.00
hadoop	13.48	4.80	0.01
SQL	12.85	4.54	0.00
optimization	11.84	5.60	0.03
big data	10.10	4.75	0.03
linux	10.02	8.67	0.25
R	7.13	4.47	0.11
matlab	5.21	7.60	0.49
SAS	3.03	5.49	0.58
NLP	0.06	9.37	0.99
oracle	-1.03	9.56	0.91
research	-1.73	4.60	0.71
software development	-4.36	8.07	0.59
UX	-6.82	14.71	0.64
stata	-6.82	14.71	0.64
data management	-9.80	7.43	0.19
ruby	-10.15	12.77	0.43
tableau	-11.51	6.10	0.06
spss	-15.11	8.59	0.08
excel	-27.63	6.02	0.00

Job Skills and Estimated Salary

Table 3 shows results from individual linear regression models, looking at the difference in estimated salary for data science positions listing vs. not listing the individual skill, after taking to account (adjusting) the differences in salary by education previously mentioned. The 3 skills associated with the largest increase in estimated salary are machine learning, C, and python, and are associated with a $27,000, $25,900, and $18,400 increase in estimated salary, respectively.

Conclusion

There is no standard or typical set of requirements for data scientist in New York City, neither for education, nor technical and analytical skills. In general, having more advanced degrees, i.e., masters and PhD, as compared to bachelors degrees, contribute to an increase in estimated salary. The analytical and technical skills desired from data scientist were associated with education, where skills related to statistics were seen more often with higher levels of education. Finally, some skills contributed as much as $27,000 to estimated salary beyond any differences in education. It is important for students, and new and experienced professionals focused in data science, to understand the skills desired from employers, and what sort of payoff one might expect from acquiring these skills.

Supplemental Code

Section 0- installing and loading packages

The authors acknowledge contributions to the slack group by Shannon Wongvibulsin who wrote the original code from which the code in section 0 was based on.

```
packages <- c("rvest", "stringr", "tidytext", "vcd", "xtable",
    "ggplot2", "crayon", "huxtable", "magrittr", "bibtex")

for (i in packages) {
    if (!require(i, character.only = T, quietly = T, warn.conflicts = F)) {
        install.packages(i, repos = "http://cran.us.r-project.org")
    }
    require(i, character.only = T, quietly = T, warn.conflicts = F)
}

library(rvest)
library(tidytext)
library(vcd)
library(xtable)
library(ggplot2)
library(crayon)
library(huxtable)
library(stringr)
library(magrittr)
library(bibtex)
```

Section 1- code used to scrape glassdoor

The authors acknowledge contributions by Stephen Cristiano who wrote the original code from which the code in section 1 was based on.

```
# this is the code I used to scrape glassdoor

## Used some of stephens code from lab for help, need to
## figure out how to cite? Load Packages


## Urls for data scientist positions on glassdoor note for
## some reason glassdoor does not let you go past job post
## page 33?



url <- paste0("https://www.glassdoor.com/Job/new-york-data-scientist-jobs-SRCH_IL.0,8_IC1132348_KO9,23_I
    1, ".htm")
urls <- paste0("https://www.glassdoor.com/Job/new-york-data-scientist-jobs-SRCH_IL.0,8_IC1132348_KO9,23
    1:33, ".htm")

df1 <- data.frame()


## First we will get data from summary data for each post from
```

```
## the search result urls



## creates fields which are the individual job posts from
## glassdoor

for (i in 1:33) {
    paste(i)
    download.file(urls[i], destfile = "scrapedpage.html", quiet = TRUE)
    fields <- read_html("scrapedpage.html") %>% html_nodes(xpath = "//*[contains(concat( \" \", @class,

    ## creates job title which are the individual job posts
    title <- fields %>% html_nodes(".flexbox .jobLink") %>% html_text() %>%
        trimws()

    ## gets salaries and replaces not listed when salary
    ## information is missing
    salaries <- sapply(fields, function(x) {
        tmp <- html_nodes(x, ".small") %>% html_text()
        ifelse(length(tmp) == 0, "Not listed", trimws(tmp))
    })
    salaries <- trimws(gsub("(Glassdoor est.)", "", salaries))

    ## gets job location


    locations <- fields %>% html_nodes(".loc") %>% html_text() %>%
        trimws()


    ## gets emplyer, employer id and job id

    ## can;t figure out how to get gsub or strreplace to recognize
    ## the - to remove the city name
    employer <- fields %>% html_nodes(".empLoc") %>% html_text() %>%
        trimws()

    employer_id <- fields %>% html_attr("data-emp-id")

    job_id <- fields %>% html_attr("data-id")


    ## create and append data frame with job posts
    df1a <- cbind(job_id, employer_id, title, salaries, locations,
        employer)

    df1 <- rbind(df1, df1a)
}

save(df1, file = "glassdoor_df1.r")
```

```
## Now we will get data from accessing the websites for the
## individual job posts using



## creates job urls from job id, note if this part breaks,
## just replace the first part of the past 0 with another link

df1$job.urls <- paste0("https://www.glassdoor.com/job-listing/data-scientist-emc-research-JV_IC1145845_
    df1$job_id)


## gets the raw job description, days posted, and employed
## from the job posts

df1$job_desc_raw <- ""
df1$employer2 <- ""
# used to compare null results in loop
a <- character(0)
n <- nrow(df1)
for (i in 1:n) {

    Sys.sleep(1)

    download.file(df1[i, ]$job.urls, destfile = "scrapedpage.html",
        quiet = TRUE)

    website <- read_html("scrapedpage.html")

    df1[i, ]$job_desc_raw <- (if (identical(a, (website %>% html_nodes("#JobDescContainer") %>%
        html_text()))) {
        "NO DESCRIPTION LISTED"
    } else {
        website %>% html_nodes("#JobDescContainer") %>% html_text()
    })

    df1[i, ]$employer2 <- (if (identical(a, (website %>% html_nodes(".padRtSm") %>%
        html_text()))) {

        "NO EMPLOYER LISTED"
    } else {
        website %>% html_nodes(".padRtSm") %>% html_text()
    })
}

saveRDS(df1, file = "glassdoor_df1")


## Finally, we will get data from acccessing the employers
## glassdoor page


## creates employer urls from job id, note if this part
```

```r
## breaks, just replace the first part of the past 0 with
## another link


unique_emp_ids <- unlist(unique(df1$employer_id[df1$employer_id !=
    0]))

emp.urls <- paste0("https://www.glassdoor.com/Overview/Working-at-ID-Analytics-EI_IE",
    unique_emp_ids, ".11,23.htm")

df2 <- data.frame(unique_emp_ids, emp.urls)

## gets empoyer description text

n <- nrow(df2)

df2$emp_desc_raw <- ""

# df2<-df2[-c(387),]

for (i in 1:n) {

    Sys.sleep(1)

    download.file(as.character(df2$emp.urls[i]), destfile = "scrapedpage.html",
        quiet = TRUE)

    website <- read_html("scrapedpage.html")

    df2[i, ]$emp_desc_raw <- (if (identical(a, (website %>% html_nodes("#EmpBasicInfo") %>%
        html_text()))) {
        "NO DESCRIPTION LISTED"
    } else {
        website %>% html_nodes("#EmpBasicInfo") %>% html_text()
    })

}

saveRDS(df2, file = "glassdoor_df2")



## merge job and employer datasets


colnames(df2)[1] <- "employer_id"

glassdoor_df <- merge(x = df1, y = df2, by = "employer_id", all.x = TRUE)

saveRDS(glassdoor_df, file = "glassdoor_df")
```

Section 2-Code used to clean text data

```
## This is the code i used to extract data from the raw job
## and employer description



# Load glass door dataset


data <- readRDS("glassdoor_df")

n <- nrow(data)

# Create structured fields from raw job descriptions



bad_txt <- c("â", "-", ",", "/")
data$job_desc1 <- gsub(paste(bad_txt, collapse = "|"), " ", data$job_desc_raw)
data$job_desc2 <- gsub("([[:lower:]])([[:upper:]])", "\\1 \\2",
    data$job_desc1)
data$job_desc2 <- gsub("([[:upper:]][[:upper:]][[:upper:]])([[:upper:]][[:lower:]])",
    "\\1 \\2", data$job_desc2)
data$job_desc3 <- gsub("[(]", " ", data$job_desc2)
data$job_desc3 <- gsub("[)]", " ", data$job_desc3)


bachelor <- c("\\bbachelor\\b", "\\bbachelors\\b", "\\bundergraduate\\b",
    "\\bbs\\b", "\\bb.s.\\b", "\\bb.s\\b")
master <- c("\\bmaster\\b", "\\bmasters\\b", "graduate degree",
    "ms degree", "ms in", "m.s. in", "m.s in")
phd <- c("phd", "doctorate", "\\bph\\b")
mba <- c("m.b.a", "\\mba\\b")
stats <- c("statistics", "statistical", "regression", "modelling")




for (i in 1:n) {

    data$python[i] <- any(grepl("\\bpython\\b", data$job_desc3[i],
        ignore.case = TRUE))
    data$ml[i] <- any(grepl("machine learning", data$job_desc3[i],
        ignore.case = TRUE))
    data$opt[i] <- any(grepl("optimization", data$job_desc3[i],
        ignore.case = TRUE))
    data$stats[i] <- any(grepl("statistic", data$job_desc3[i],
        ignore.case = TRUE))
    data$risk[i] <- any(grepl("risk", data$job_desc3[i], ignore.case = TRUE))
    data$UX[i] <- any(grepl("UX", data$job_desc3[i], ignore.case = FALSE))
    data$bd[i] <- any(grepl("big data", data$job_desc3[i], ignore.case = TRUE))
    data$dm[i] <- any(grepl("data management", data$job_desc3[i],
```

```
    ignore.case = TRUE))
  data$pharma[i] <- any(grepl("pharmaceutical", data$job_desc3[i],
    ignore.case = TRUE))
  data$fs[i] <- any(grepl("financial services", data$job_desc3[i],
    ignore.case = TRUE))
  data$sd[i] <- any(grepl("software development", data$job_desc3[i],
    ignore.case = TRUE))
  data$program[i] <- any(grepl("programming", data$job_desc3[i],
    ignore.case = TRUE))
  data$research[i] <- any(grepl("research", data$job_desc3[i],
    ignore.case = TRUE))
  data$R[i] <- any(grepl("\\bR\\b", data$job_desc3[i], ignore.case = TRUE))
  data$SAS[i] <- any(grepl("\\bSAS\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$C[i] <- any(grepl("\\bC+\\b", data$job_desc3[i], ignore.case = TRUE))
  data$stata[i] <- any(grepl("\\bstata\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$SQL[i] <- any(grepl("\\bsql\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$excel[i] <- any(grepl("\\bexcel\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$tableau[i] <- any(grepl("\\btableau\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$spss[i] <- any(grepl("\\bspss\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$java[i] <- any(grepl("\\bjava\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$linux[i] <- any(grepl("\\blinux\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$matlab[i] <- any(grepl("\\bmatlab\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$NLP[i] <- any(grepl("\\bNLP\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$hadoop[i] <- any(grepl("\\bhadoop\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$ruby[i] <- any(grepl("\\bruby\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$oracle[i] <- any(grepl("\\boracle\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$sas[i] <- any(grepl("\\bsas\\b", data$job_desc3[i],
    ignore.case = TRUE))
  data$bs[i] <- any(grepl(paste(bachelor, collapse = "|"),
    data$job_desc3[i], ignore.case = TRUE))
  data$bs2[i] <- any(grepl("bachelor", data$job_desc3[i], ignore.case = TRUE))
  data$masters[i] <- any(grepl(paste(master, collapse = "|"),
    data$job_desc3[i], ignore.case = TRUE))
  data$phd[i] <- any(grepl(paste(phd, collapse = "|"), data$job_desc3[i],
    ignore.case = TRUE))
  data$mba[i] <- any(grepl(paste(mba, collapse = "|"), data$job_desc3[i],
    ignore.case = TRUE))
  data$stats[i] <- any(grepl(paste(stats, collapse = "|"),
    data$job_desc3[i], ignore.case = TRUE))
  data$degree[i] <- any(grepl("\\bdegree\\b", data$job_desc3[i],
```

```
        ignore.case = TRUE))

}

# data$skills<-
# data$degree+data$phd+data$masters+data$phd+data$bs2+data$bs+data$sas+data$oracle+data$ruby+data$hadoo
# data$java+ data$spss +data$spss +data$tableau+
# data$excel+data$SQL+data$stata + data$C + data$SAS +data$R+
# data$python

data$skills2 <- data$sas + data$oracle + data$ruby + data$hadoop +
    data$NLP + data$matlab + data$linux + data$java + data$spss +
    data$spss + data$tableau + data$excel + data$SQL + data$stata +
    data$C + data$SAS + data$R + data$python + data$ml + data$opt +
    data$bd + data$research + data$stats + data$dm + data$risk +
    data$fs + data$program + data$pharma + data$sd + +data$UX

data$education <- data$masters + data$phd + data$bs


# table(data$skills2) table(data$education)

# saveRDS(data, file='glassdoor_df_cleaned')


# data$job_id[data$skills2==0]
# strsplit(data$job_desc3[data$job_id=='2426165858'],' ')


# Create structured fields from raw employer descriptions



data$emp_desc1 <- gsub(paste(bad_txt, collapse = "|"), " ", data$emp_desc_raw)
data$emp_desc2 <- gsub("([[:lower:]])([[:upper:]])", "\\1 \\2",
    data$emp_desc1)
data$emp_desc2 <- gsub("([[:lower:]])([[:digit:]])", "\\1 \\2",
    data$emp_desc2)
data$emp_desc2 <- gsub("([[:digit:]])([[:upper:]])", "\\1 \\2",
    data$emp_desc2)

sub(".*years experience *(.*?)", "\\1", data$job_desc3[1], ignore.case = TRUE)

data$job_desc3[1]

for (i in 1:n) {

    data$founded[i] <- any(grepl("\\bfounded\\b", data$emp_desc2[i],
        ignore.case = TRUE))
    data$industry[i] <- sub(".*Industry *(.*?) *Revenue.*", "\\1",
        data$emp_desc2[i], ignore.case = TRUE)
    data$revenue[i] <- any(grepl("\\brevenue\\b", data$emp_desc2[i],
        ignore.case = TRUE))
```

```
}



for (i in 1:n) {

    data$salary_low[i] <- sub(".*[$] *(.*?) *k[-].*", "\\1",
        data$salaries[i], ignore.case = TRUE)
    data$salary_high[i] <- sub(".*[$] *(.*?) *k[-].*", "\\1",
        data$salaries[i], ignore.case = TRUE)

}


## get average salary

data$salary_low[data$salary_low == "Not listed"] <- ""
data$salary_high[data$salary_high == "Not listed"] <- ""


data$salary_average <- as.numeric(data$salary_low) + as.numeric(data$salary_high)



saveRDS(data, file = "glassdoor_df_cleaned")
```

References

Burning Glass Technologies. 2017. "The Quant Crunch: How the Demand for Data Science Skills Is Disrupting the Job Market." http://burning-glass.com/wp-content/uploads/The_Quant_Crunch.pdf.

Glassdoor. 2017. "What Are Salary Estimates in Job Listings?" http://help.glassdoor.com/article/What-are-Salary-Estimates-in-Job-Listings.

Pierson, Lillian. 2017. "Top Data Science Skills in 2017: Identify Where to Work and the Skills to Land You There." http://www.data-mania.com/blog/top-data-science-skills-in-2017/.

Ramel, David. 2016. "What Are the Most-Wanted Data Science Skills for 2016?" https://adtmag.com/articles/2016/01/08/data-science-skills.aspx.

Silge, Julia, and David Robinson. 2016. "Tidytext: Text Mining and Analysis Using Tidy Data Principles in R." *JOSS* 1 (3). The Open Journal. doi:10.21105/joss.00037.

Strauss, Karsten. 2017. "Becoming a Data Scientist: The Skills That Can Make You the Most Money." https://www.forbes.com/sites/karstenstrauss/2017/09/21/becoming-a-data-scientist-the-skills-that-can-make-you-the-most-money/#2c9c8b2e634f.

Wickham, Hadley. 2016. *Rvest: Easily Harvest (Scrape) Web Pages*. https://CRAN.R-project.org/package=rvest.

———. 2017. *Stringr: Simple, Consistent Wrappers for Common String Operations*. https://CRAN.R-project.org/package=stringr.

Wikipedia, the free encyclopedia. 2017. "Data Science." https://en.wikipedia.org/wiki/Data_science.