DATA 607: Project 3

Michael Munguia, Keshaw Sahay, James Williams & Chunsan Yip

3/22/2020

Introduction

The project goal was to gather and explore data revolving around the necessary job skills for a data scientist in today's job market. As a group, we discussed potential avenues for data collection via slack and based on availabilities and time constraints, executed the most readily accessible strategies. Unlike a real-world work scenario, we do not have most of our waking hours to dedicate to this, so we had to operate quickly and efficiently to deliver.

Overall Process

There was no single process for gathering the data, but we focused on three main sources of data: job postings from NYC's various government agencies, articles written about the important skills needed by a data scientist and survey responses regarding data science working skills. The process for gathering data was two fold: mining data files for the job posts and survey results along with manual collection for the articles.

Data gathering & exploration

NYC Open Data: Job Postings

Function writing

Creating a few custom functions, the process for collecting and processing data from job descriptions posted onto the city's job post mechanism boiled down to reading a CSV and tokenizing the two columns dictating required and preferred skills such that we break each text document down into its constituent words. The text needed to be cleaned to standardize the words across descriptions such that artificially distinct words were not created as a result of things like special characters, punctuation, whitespace or capitalization. The cleaning process is managed by the `clean_description` function and tokenization was handled by the `tidytext::unnest_tokens` function. Once cleaned and tokenized, the text data was passed to the `word_frequency` function to remove any stop words, including a special list created by iteratively evaluating the results, and ultimately generate counts for each word in the overall corpus of description texts.

```r
library(tidytext)

clean_description <- function(text) {
  text <- text %>%
    iconv("", "ASCII", "byte") %>%
    str_remove_all("<.+>|\t|\\d|[:punct:]") %>%
    str_trim() %>%
    str_to_lower()

  return(text)
}

word_frequency <- function(df, addl_stopwords = NA) {
  stop_words <- get_stopwords()
  if (is.character(addl_stopwords)) {
    stop_words <- bind_rows(stop_words,
                            tibble(word = addl_stopwords,
                                   lexicon = rep("user defined",
                                                 length(addl_stopwords)
                                   )
                            )
    )
  }

  df %>%
    anti_join(stop_words, by = "word") %>%
    count(word, sort = TRUE)
}

unnest_wrapper <- function(df, target_var, word_var = "word") {
  df %>%
    unnest_tokens_(word_var, target_var) %>%
    select(title_id, category_id, word)
}


word_plot <- function(df, addl_stopwords = NA, top_n = 10,
                      title_lab = "Job Posting",
                      x_lab = "Words in Text", y_lab = "Frequency",
                      fill_color = "gray") {
  top_df <- word_frequency(df, addl_stopwords) %>% head(top_n)
  word_order <- top_df$word
  top_df <- top_df %>% mutate_at("word", factor, ordered = TRUE, levels = word_order)

  result_plot <- ggplot(top_df, aes(word, n)) +
    geom_col(fill = fill_color) +
    labs(title = title_lab, x = x_lab, y = y_lab)

  if (top_n < 11) {
    return(result_plot)
  } else {
    return(result_plot + coord_flip())
  }
}

# These are additional stopwords that appear to be noise in the greater scheme
# of what we're trying to do (i.e. extract meaningful words from job postings)
jobpost_stopwords <- c("experience", "years", "accredited", "college", "degree",
                       "field", "equivalent", "appropriate", "education",
                       "specialization", "described", "year", "ability",
                       "skills", "data", "strong", "work", "working", "well",
                       "must", "candidate", "candidates", "university", "least",
                       "community", "fulltime", "school", "andor", "one", "two",
                       "three", "four", "five", "six", "seven", "eight", "nine",
                       "ten", "high", "level", "area", "related", "however",
                       "satisfactory", "assignment", "public", "duties", "social",
                       "activities", "centered", "responsible", "diploma", "period",
                       "probationary", "substituted", "accrediting", "graduation",
                       "recognized", "approved", "fouryear", "note", "required",
                       "capacity", "department", "development", "organization",
                       "states", "credits", "ii", "appointed", "appointments",
                       "program", "position", "can", "combination", "may",
                       "subject", "including", "semester", "appointment", "basis",
                       "graduate", "acquired", "made", "major", "requirements",
                       "use", "agencies", "areas", "bodies", "chea", "council",
                       "estate", "following", "hire", "june", "meeting", "past",
                       "real", "promotion", "primarily", "plus", "us", "using",
                       "user", "valid", "additional", "august", "b", "c", "base",
                       "eligible", "employee", "employment", "end", "fields",
                       "fifteen", "general", "held", "paid", "possess",
                       "prospective", "qualification", "recent", "winterspring",
                       "within", "without", "yearforyear", "oror", "post", "agency",
                       "city", "minimum", "months", "new", "york", "large", "highly",
                       "desired", "able", "interest", "part", "etc", "strongly",
                       "time", "prefer", "preference", "nyc", "environment",
                       "excellent", "familiarity", "demonstrated", "computer",
                       "administration", "administrative", "tasks", "student")
```

Data wrangling

After loading the `tidytext` library and defining my custom functions, the data is loaded and run through the process described above. Additionally, unique identifiers are created for both the job titles and categories listed in the job posts.

```
jobs <- read_csv(nyc_jobs) %>%
  select("title" = `Business Title`, "category" = `Job Category`,
         "requirements" = `Minimum Qual Requirements`,
         "preferences" = `Preferred Skills`)

data_jobs <- jobs %>%
  filter(str_detect(title, regex("data",ignore_case = TRUE)) &
           str_detect(title, regex("analyst|scientist|engineer", ignore_case = TRUE))) %>%
  mutate_at(c("requirements", "preferences"), clean_description)

title_ids <- data_jobs %>% group_indices(title)
category_ids <- data_jobs %>% group_indices(category)

data_jobs$title_id <- title_ids
data_jobs$category_id <- category_ids
```

Putting it all together

With unique identifiers in hand, separate dataframes are established for later normalized SQL table creation. In essence: `title_table` for job titles, `category_table` for departmental categories and `job_requirements` for required skills and `job_preferences` for preferred skills. The primary key linking these all will be the unique `title_id` and/or `category_id` . This leaves room open later on to further standardize the job titles such that departmental information could be extracted from the titles themselves if we eventually wanted to dissect job skills for data analysts versus data scientists, senior positions versus junior positions, etc.

```
(title_table <- data_jobs %>% distinct(title_id, title))
```

```
## # A tibble: 14 x 2
##    title_id title
##       <int> <chr>
##  1        3 Data Analyst
##  2       14 Senior Data Analyst M/WBE, Enterprise Data Services
##  3        7 Data Scientist
##  4        4 Data Analyst, Bureau of HIV
##  5       10 NYC Census 2020 - Community Outreach Data Analyst
##  6        6 Data Analyst, PNA
##  7       11 Policy Analyst - Data
##  8        8 Data Support Analyst
##  9       13 Program Data Analyst
## 10        1 2020-BWS-014-Reservoir Ops Associate Data Analyst Intern
## 11       12 Policy Analyst  - Data and Technology
## 12        5 DATA ANALYST, BUSINESS OPERATIONS
## 13        9 HRIS Data Analyst
## 14        2 Business and Data Analyst Manager
```

```
(category_table <- data_jobs %>% distinct(category_id, category))
```

```
## # A tibble: 13 x 2
##    category_id category
##          <int> <chr>
##  1           5 Finance, Accounting, & Procurement Technology, Data & Innovation~
##  2          11 Technology, Data & Innovation
##  3           9 Policy, Research & Analysis Public Safety, Inspections, & Enforc~
##  4           8 Policy, Research & Analysis
##  5           6 Health
##  6           1 Constituent Services & Community Programs
##  7          12 Technology, Data & Innovation Legal Affairs Policy, Research & A~
##  8           3 Engineering, Architecture, & Planning
##  9           2 Constituent Services & Community Programs Technology, Data & Inn~
## 10          13 Technology, Data & Innovation Policy, Research & Analysis
## 11          10 Public Safety, Inspections, & Enforcement
## 12           7 Legal Affairs Policy, Research & Analysis
## 13           4 Finance, Accounting, & Procurement
```

```
data_jobs <- data_jobs %>% distinct(title_id, category_id, requirements, preferences)


(job_requirements <- data_jobs %>% unnest_wrapper("requirements"))
```

```
## # A tibble: 2,948 x 3
##    title_id category_id word
##       <int>       <int> <chr>
##  1        3           5 a
##  2        3           5 masters
##  3        3           5 degree
##  4        3           5 from
##  5        3           5 an
##  6        3           5 accredited
##  7        3           5 college
##  8        3           5 or
##  9        3           5 university
## 10        3           5 accredited
## # ... with 2,938 more rows
```

```
(job_preferences <- data_jobs %>% unnest_wrapper("preferences"))
```

```
## # A tibble: 1,238 x 3
##    title_id category_id word
##       <int>       <int> <chr>
##  1        3           5 experience
##  2        3           5 position
##  3        3           5 qualifications
##  4        3           5 commitment
##  5        3           5 applicants
##  6        3           5 must
##  7        3           5 be
##  8        3           5 in
##  9        3           5 the
## 10        3           5 title
## # ... with 1,228 more rows
```

Database interlude I

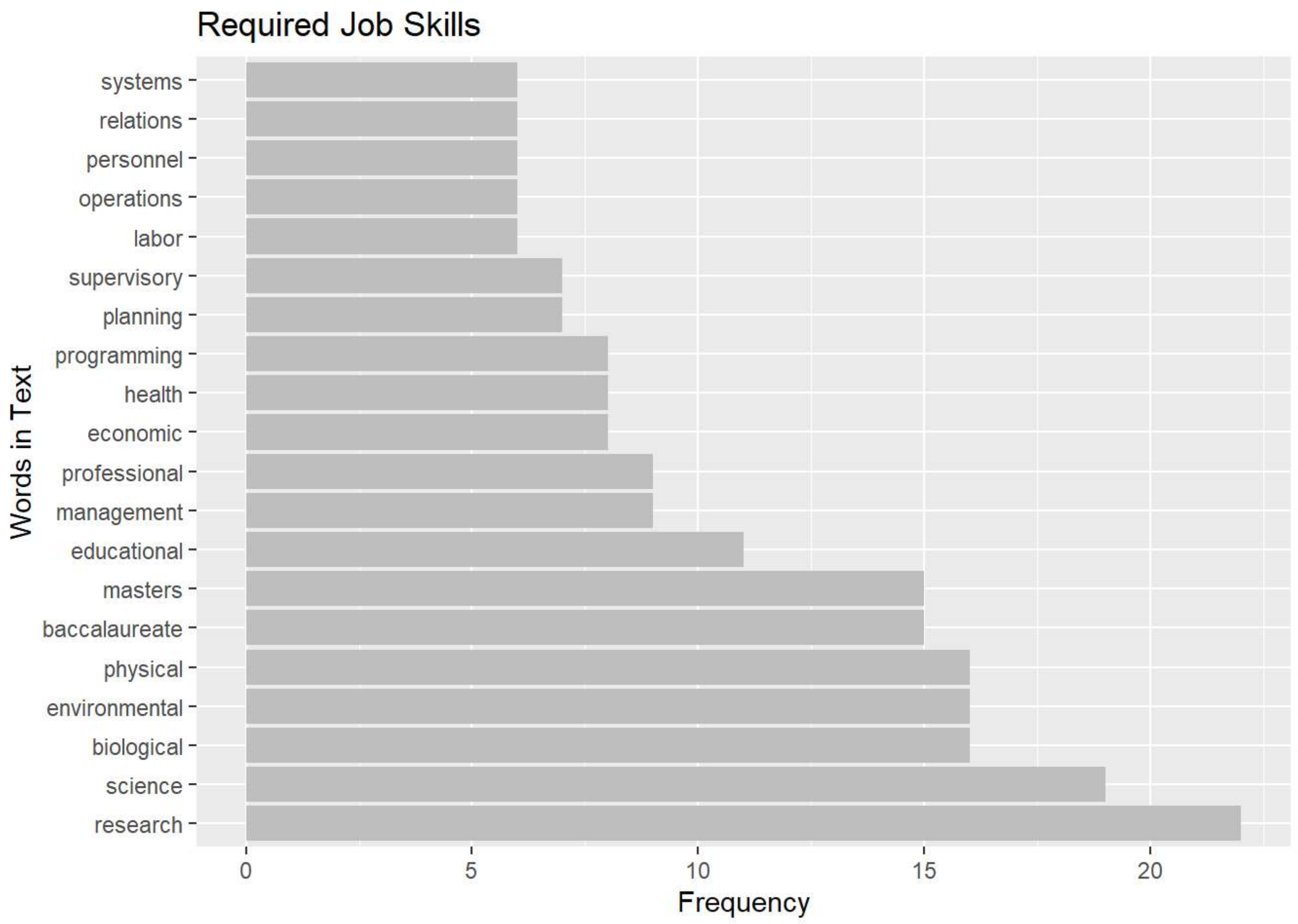
Writing the tables above as CSV Files, normalized tables were created and populated via the PGAdmin 4 GUI. Alternatively, a backup file is included in this repository. To load the data back into R, a process like below could be followed. For reproducibility, I maintained the dataframes created thus far, but if one were to instantiate the database via the backup file and load accordingly the results will be the same.:

```
library(DBI)
library(odbc)

connection <- dbConnect(RPostgres::Postgres(), dbname = params$db_name,
                        host = params$host, user = params$user, password = params$pwd)

read_db_tbl <- function(conx, select_statement) {
  as.tbl(dbGetQuery(conn = conx, statement = select_statement))
}

title_table <- read_db_tbl(connection, "SELECT * FROM title_table")
category_table <- read_db_tbl(connection, "SELECT * FROM category_table")

job_requirements <- read_db_tbl(connection, "SELECT * FROM job_requirements")
job_preferences <- read_db_tbl(connection, "SELECT * FROM job_preferences")
```

Analysis

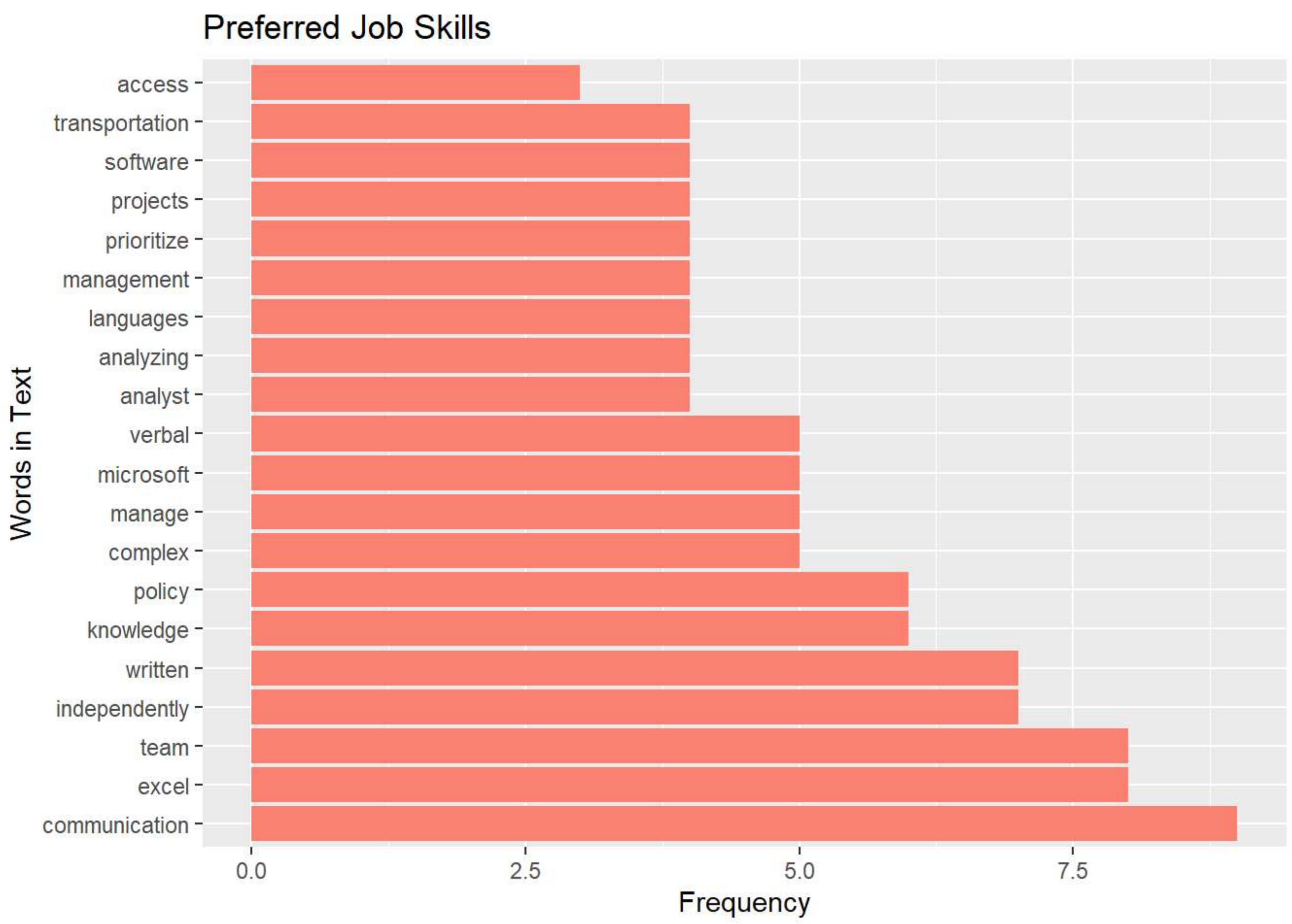
The top twenty most frequent words that appear in required job skills are below. More work is required here to adequately parse descriptions, but we can get a handle on the fact that most required skills revolve around having a more scientific background - words like research, biological, environmental and both masters and baccalaureate round out the top ten along with words like management and professional. We can glean from this that, at least as far as the city of New York is concerned, they are looking to hire data specialists with a higher degree of education than not. A research background in one of the sciences may give a potential applicant an edge over the competition.

```
word_plot(job_requirements, jobpost_stopwords, 20, title = "Required Job Skills", fill_color =
"gray")
```



Likewise, we can see preferred job skills below. It is interesting to me that no language indicating any specific tools appears above, but we do see plenty of them below. Surprisingly, elements of the Microsoft Office suite are important: we see Excel appearing prominently as second most referenced word in preferred skills for these jobs. It is tied with a reference to “team”, likely indicating being able to work on a team is highly valued along with the top preferred skill: communication. While a higher educational background may get one in the door, collaborative skills seem to be much more important. Certainly, we see references to analysis, software and (programming) languages but half of the top twenty refer to some kind of human-facing skill that we might not immediately think of in the wizardly world of data.

```
word_plot(job_preferences, jobpost_stopwords, 20, title = "Preferred Job Skills",
          fill_color = "salmon")
```



Article Skill Lists

Another avenue was collecting skills described by online articles set to highlight the most important data science skills. In this case, 10 articles were reviewed and data was manually collected into a CSV file.

Data loading / Database Interlude II

Again, for ease of reproducibility, the data will be read from a CSV file. In the comments, however, would be how to load the data given the same connection as before. The rest of the process remains the same.

```
skills <- read_csv(vs_file, skip=2)
# skills <- read_db_tbl(connection, "SELECT * FROM valued_skills")
```

Data wrangling

The data is manipulated into a workable dataframe below.

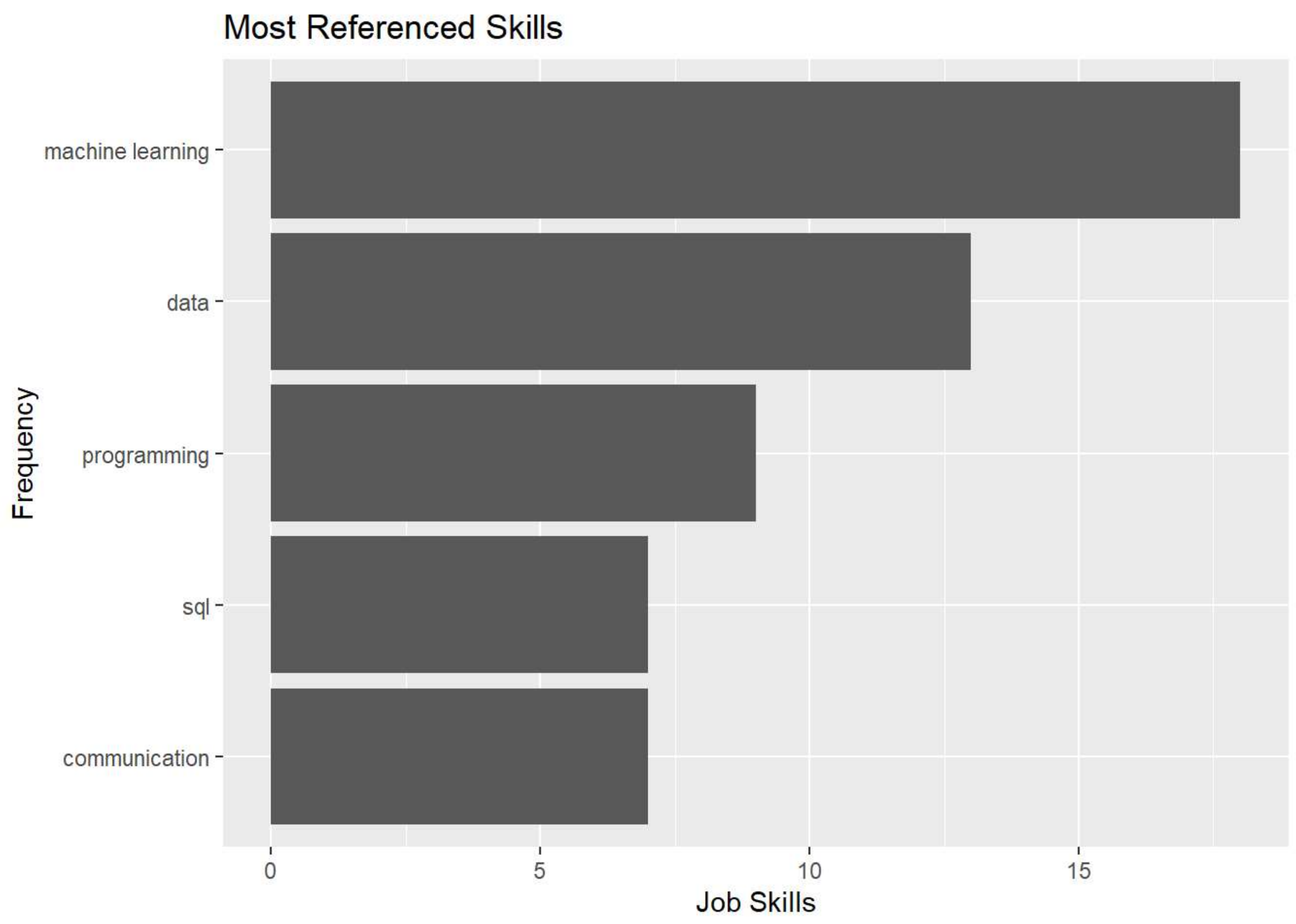
```
a<-max(str_count(skills$Skill, "\\s"))

i=1
newCol <- c(1)
for(i in 1:(a+1)) {
  newCol[i] <- as.character(i)
}

newSkills <- skills %>%
  separate(Skill, into = newCol, sep="\\s", fill="right")%>%
  gather(key="nCol", value="nSkill", 2:9) %>%
  mutate(nSkill = str_to_lower(nSkill)) %>%
  mutate(nSkill = str_replace(nSkill, "[ ,?:]", "") ) %>%
  filter(!is.na(nSkill)) %>%
  arrange(desc(nSkill))

# Machine and Learning are actually linked together after reviewing the csv file again

gskills <- newSkills %>%
  mutate(nSkill = if_else(nSkill %in% c("machine", "learning"), "machine learning", nSkill)) %>%
  group_by(nSkill)%>%
  summarise(n=n())%>%
  arrange(desc(n))

gskills
```

```
## # A tibble: 92 x 2
##    nSkill               n
##    <chr>            <int>
##  1 machine learning    18
##  2 data                13
##  3 programming          9
##  4 communication        7
##  5 sql                  7
##  6 and                  6
##  7 &                    5
##  8 python               5
##  9 r                    5
## 10 skills               5
## # ... with 82 more rows
```

Results

The top five skills should be “Data” (including data analysis, data visualization, etc.), “Machine Learning”, “Programming”, “Communication” and “SQL”.

```
ggplot(data=(gskills[1:5,])) +
  geom_col(mapping = aes(x = fct_reorder(nSkill, n), y = n)) +
  coord_flip() +
  labs(title = "Most Referenced Skills", x = "Frequency", y = "Job Skills")
```



Survey Data

Data Acquisition

The data set used in this project is Kaggle ML and Data Science Survey 2017 (https://www.kaggle.com/kaggle/kaggle-survey-2017/feed). I downloaded the multiple choice item survey results in csv format and placed it in a GitHub repo (https://github.com/keshaws/CUNY_MSDS_2020/tree/master/DATA607 (https://github.com/keshaws/CUNY_MSDS_2020/tree/master/DATA607))

Importing Multiple Choice data

```
surverydata_df <- read_csv (surverydata_link)
survey.data <- surverydata_df
#lets create a unique ID variable
surverydata_df$id <- seq.int(nrow(surverydata_df))
dim(surverydata_df)
```

```
## [1] 16716   229
```

Research Question

Which are the most values data science skills?

Understanding Features

Let’s start gaining some insights by exploring demographics features of the dataset.

```
survey.demographics <- survey.data %>%
  select(GenderSelect,Country,Age,EmploymentStatus) %>%
  filter(Country!='NA',trimws(Country)!='',Age!='NA',trimws(GenderSelect) %in% c('Male','Female'
))

# These workable results were stored to a database, and can be loaded given its restoration as f
ollows:
# survey.demographics <- read_db_tbl(connection, "SELECT * FROM survey_demographics")
```

```
survey.dem.age.plot <- survey.demographics %>%
    group_by(Age,GenderSelect) %>%
    summarise(count=n()) %>%
    arrange(desc(count))
survey.dem.plot <- survey.demographics %>%
  group_by(Age,Country,GenderSelect,EmploymentStatus) %>%
  summarise(count=n()) %>%
  arrange(desc(count))
```

```
survey.dem.gen.plot <- survey.demographics %>%
    group_by(GenderSelect) %>%
    summarise(count=n()) %>%
    arrange(desc(count))
head(survey.dem.gen.plot)
```

```
## # A tibble: 2 x 2
##   GenderSelect count
##   <chr>        <int>
## 1 Male         13416
## 2 Female        2708
```

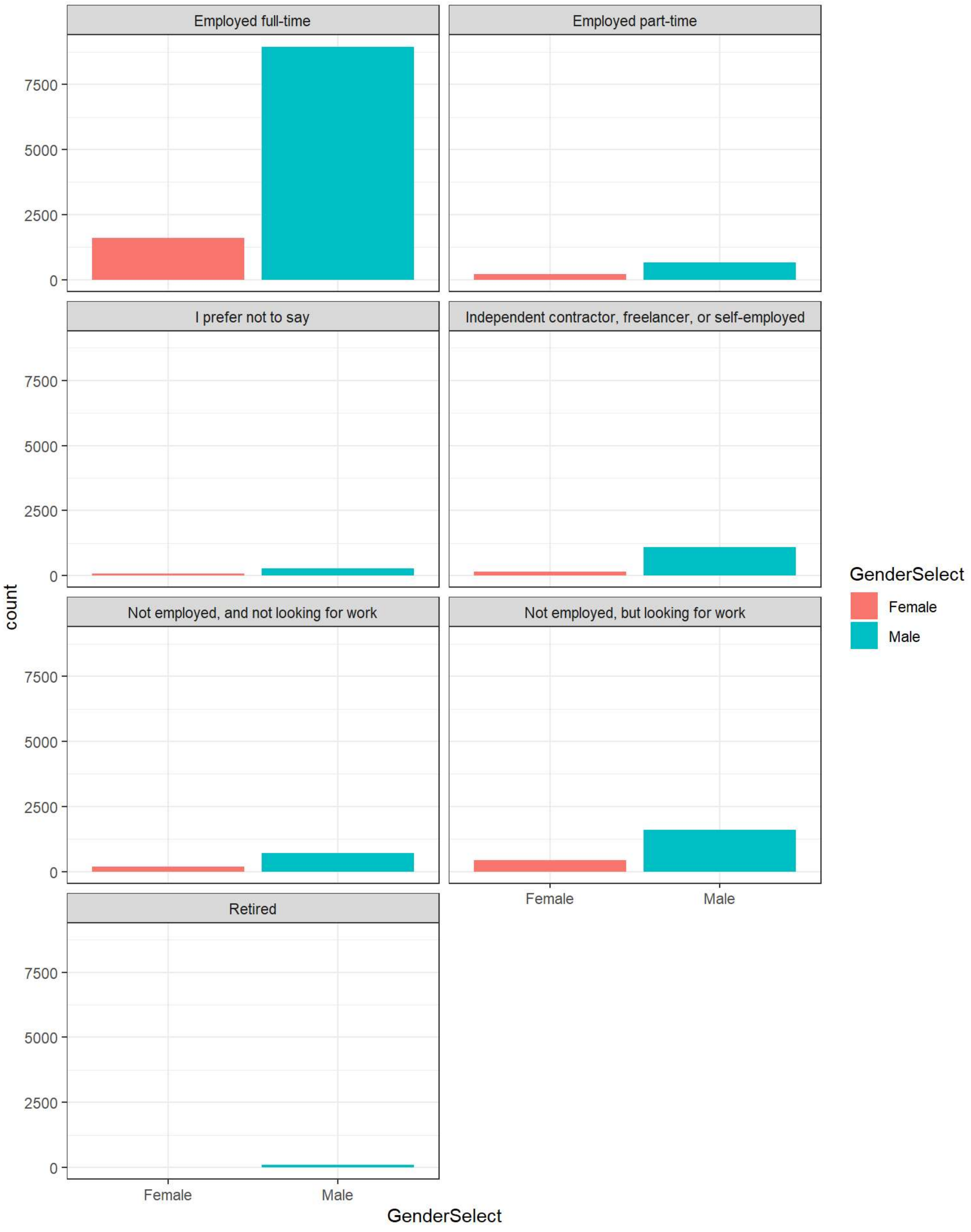
Tidying the dataset and finding the percentile for each gender group

```
survey.dem.tidy <- survey.dem.gen.plot %>%
    spread(GenderSelect,'count')
head(survey.dem.tidy)
```

```
## # A tibble: 1 x 2
##   Female  Male
##    <int> <int>
## 1   2708 13416
```

```
gender_percentile <- mutate(survey.dem.tidy,male_percent=round((Male/(Female+Male)*100),2),
                            female_percent=round((Female/(Female+Male))*100,2))
kable(gender_percentile)
```

Female	Male	male_percent	female_percent
2708	13416	83.21	16.79



There are 16716 survey respondents and we could see that there is a huge gender gap in the given dataset with over 83% are male and female makes up only ~17% of total. Also, most of respondents are full time employed followed by people who are not employed but looking for work.

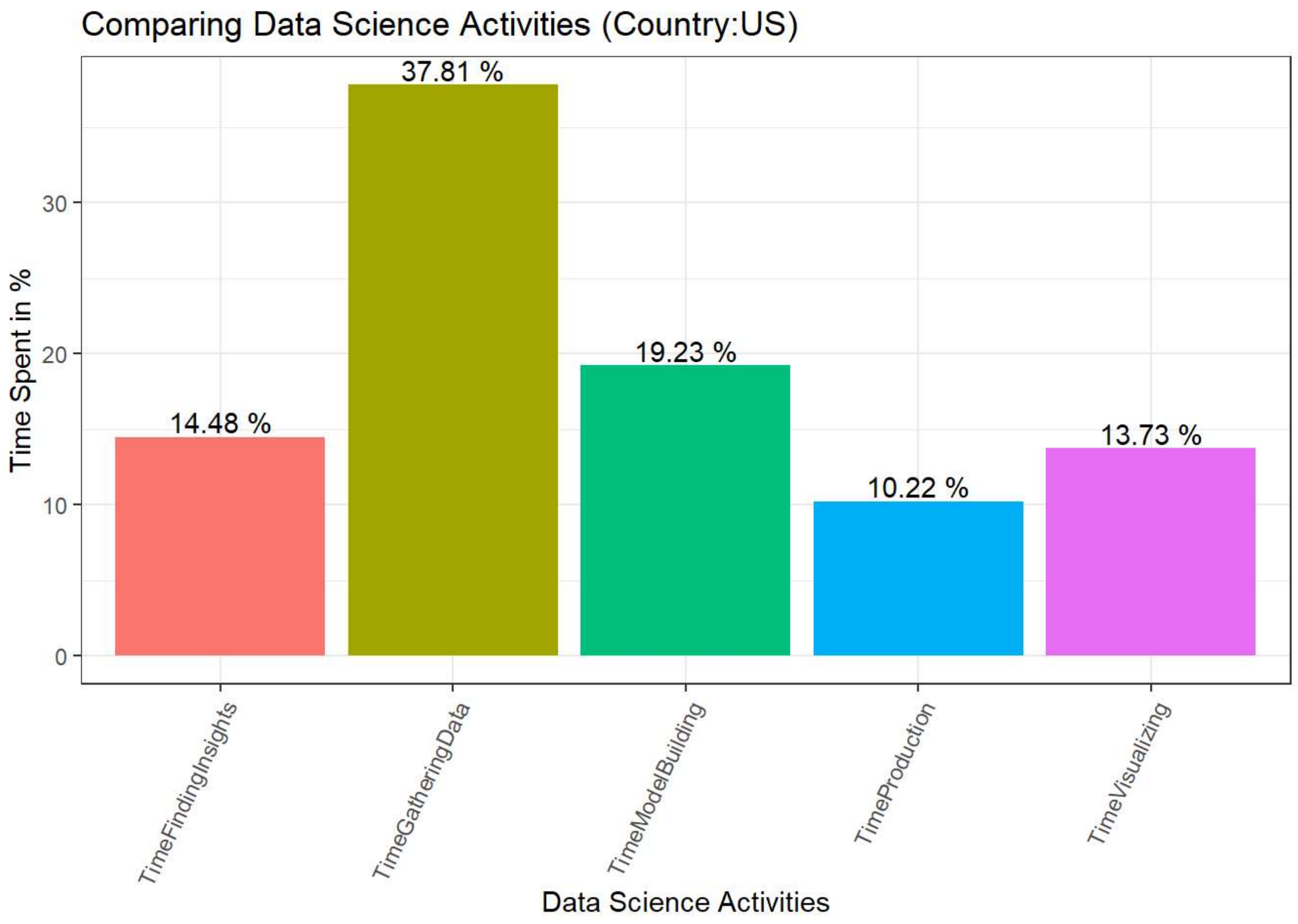
EDA

Let’s take a look at data science activity attributes: TimeGatheringData, TimeModelBuilding, TimeProduction, TimeVisualizing, TimeFindingInsights.

Data Science Activities

The US reponsdents data analysis show that gathering data is the main activitiy with higest time consumption 37.75%. The model building ranks 2nd, 19.23%, followed by time spent in finding insights and data visualization. Only 10.22% of total appears to be taken by prodcution activities.

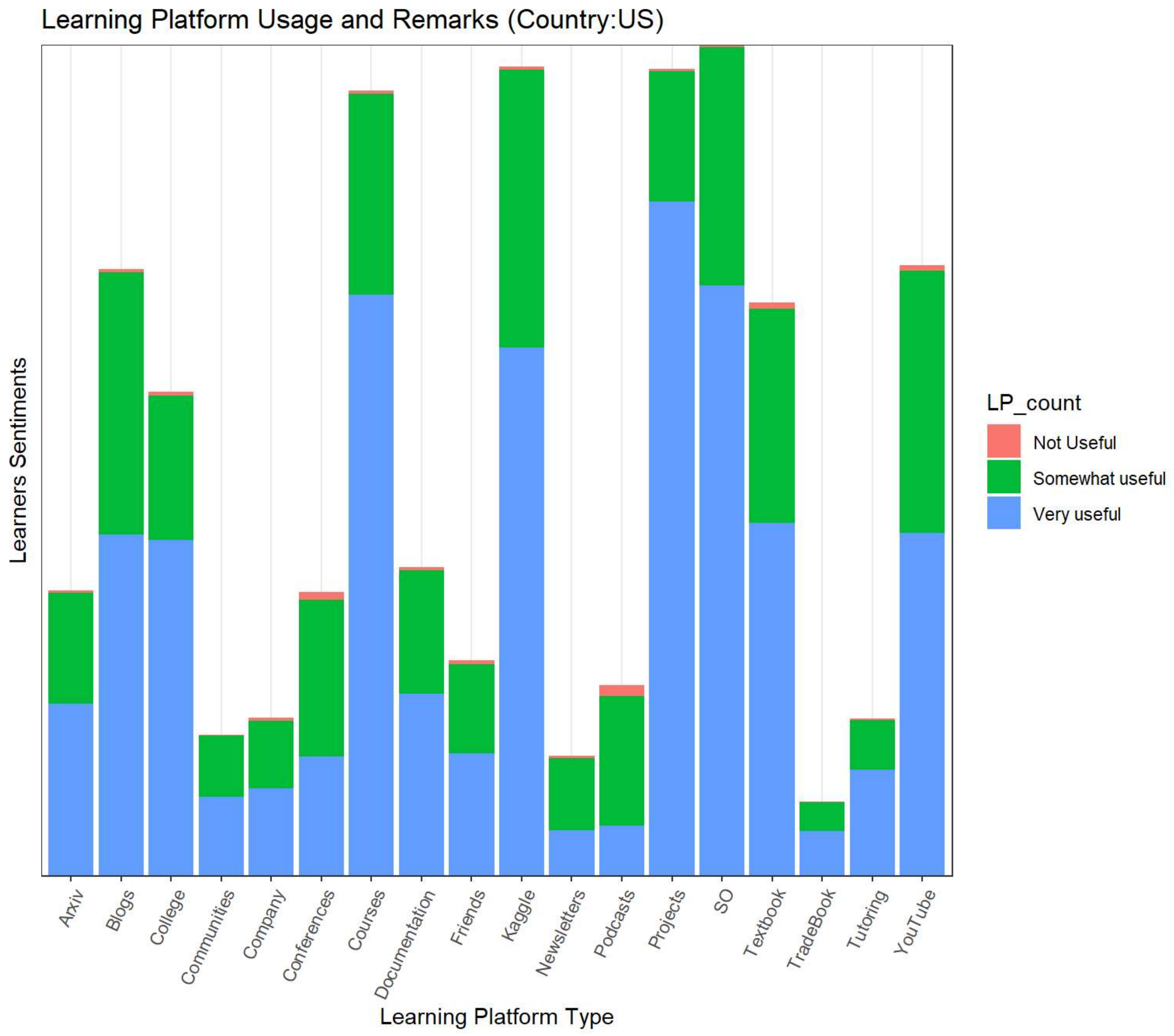
DSActivity	mean_precent
TimeGatheringData	37.81491
TimeModelBuilding	19.23414
TimeFindingInsights	14.48332
TimeVisualizing	13.72629
TimeProduction	10.22008



Learning platform

Iid	Country	EmploymentStatus	LPlatform	LP_count	LearningPlatform
1	United States	Not employed, but looking for work	LearningPlatformUsefulnessKaggle	Somewhat useful	Kaggle

Iid	Country	EmploymentStatus	LPlatform	LP_count	LearningPlatform
3	United States	Independent contractor, freelancer, or self-employed	LearningPlatformUsefulnessBlogs	Very useful	Blogs
3	United States	Independent contractor, freelancer, or self-employed	LearningPlatformUsefulnessCollege	Very useful	College
3	United States	Independent contractor, freelancer, or self-employed	LearningPlatformUsefulnessConferences	Very useful	Conferences
3	United States	Independent contractor, freelancer, or self-employed	LearningPlatformUsefulnessFriends	Very useful	Friends
3	United States	Independent contractor, freelancer, or self-employed	LearningPlatformUsefulnessDocumentation	Very useful	Documentation



The survery respondents used different learning platform and it appears that learners mostly benefited from personal projects as majority of resonse indicate it very useful. Online courses appears to be 2nd very useful only to be followed by StackOverflow and Kaggle. Blogs,textbooks and college also appear to be very userful whereas newsletters, podcasts and tradebook rank low.

Conclusion

Based on the different approaches taken by our team, we have seen a variety of different skill sets that are important for a data scientist to acquire and develop. While “soft-skills” like the ability to communicate clearly and navigate working in a collaborative environment are extremely in demand, they must be coupled with that “hard-skills” we are all familiar with: programming, data modeling, etc. In pursuit of this, there are a myriad of sources from which to draw from in becoming a more able practitioner of data science, though perhaps there is no better teacher than personal practice in the form of engaging personal projects and online learning.