H1-B Work Visa Applications 2011-2016

William M.

February 18, 2018

Investigating H1B-Visa Application Data from 2011-2016

Using a publicly available dataset documenting the total number of H1B applications (both certified and rejected) for the years 2011-2016 we hope to gain insight into the phenomenon of temporary work visas in the United States. H1B visas are awarded to foreign workers with *specialized skills* who have already been offered employment in the United States on a temporary basis (initially 3 years). The certified applications from this dataset are then considered for final approval. H1B visas are awarded by the United States Citizenship and Immigration Services and awarded on a first come, first served basis with a total cap of 85,000 work visas per year.
Source: https://www.kaggle.com/nsharan/h-1b-visa User: Sharan Naribole

Cleaning up the SOC_NAME Variable

The SOC_NAME variable in this dataset is the job category of the job being applied for based on the Standard Occupational Classification guidelines. The immediate problem in working with this data is that each case does not include the associated SOC code and that the SOC names are inconsistently recorded. For example, in the entire dataset there are 21,251 cases with job categories variously recorded as:

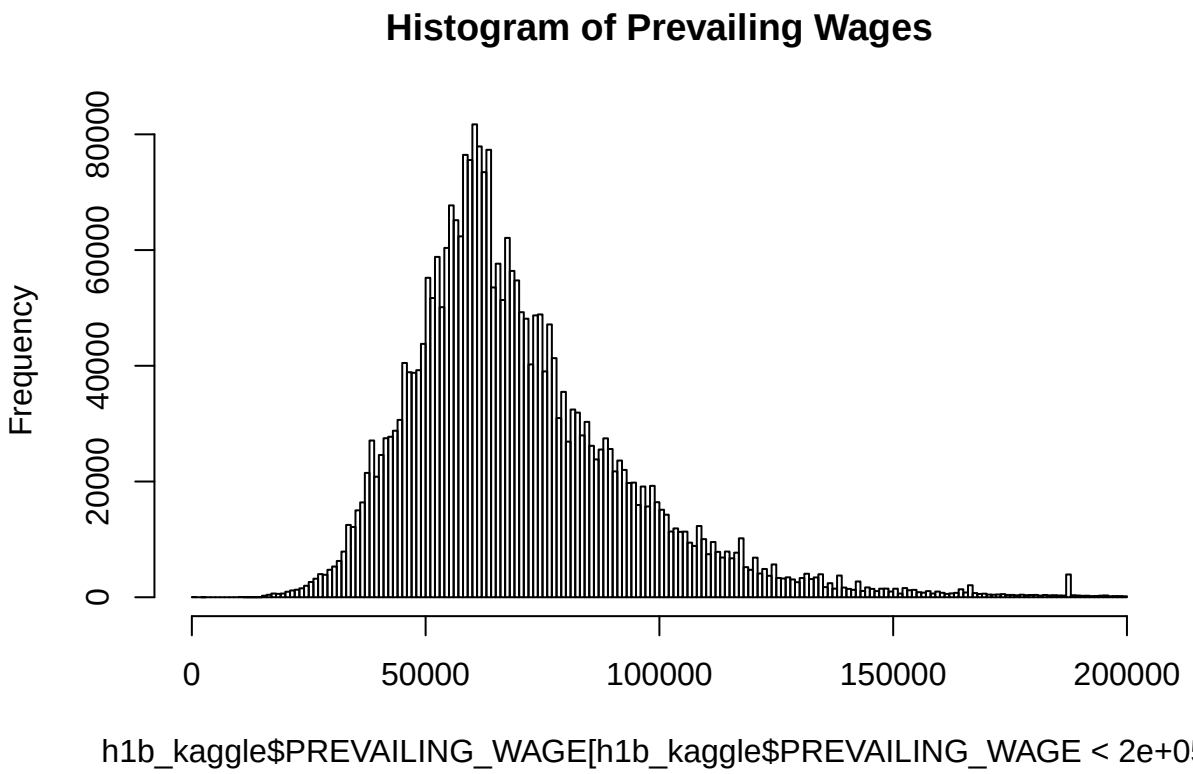
```
## [1] "BIOCHEMISTS AND BIOPHYSICISTS" "BIOCHEMISTS & BIOPHYSICISTS"
## [3] "BIOCHEMISTS OR BIOPHYSICISTS"  "BIOCHEMIST & BIOPHYSICIST"
## [5] "Biochemists and Biophysicists"
```

Clearly, all of these cases should be considered to be in the same job category.
In order to aggregate cases that clearly belong to the same SOC_NAME but contain spelling errors or abbreviations we wrote the new_dataframe R script saved in the ReduceNames.R file. In short this function calculates the adist string distance (this distance is the minimum number of additions, deletions, or substitutions needed to interchange the two strings) between every pair of unique SOC_NAME's and assigns them to a new variable called "names_class" where every SOC_NAME assigned to the same "names_class" is within a fixed distance to the other names in that class.
This way we can safely aggregate each application based on its "names_class" and assume that these aggregate cases are the same within a small margin of error.

Exploratory analysis

The distribution of prevailing wages for all jobs applied to is.

```
hist(h1b_kaggle$PREVAILING_WAGE[h1b_kaggle$PREVAILING_WAGE < 200000], breaks = 200, xlim=c(0,200000), ma
```



To make the histogram more readable we are only including wages under $200,000 which accounts for 99.65% of all wages.

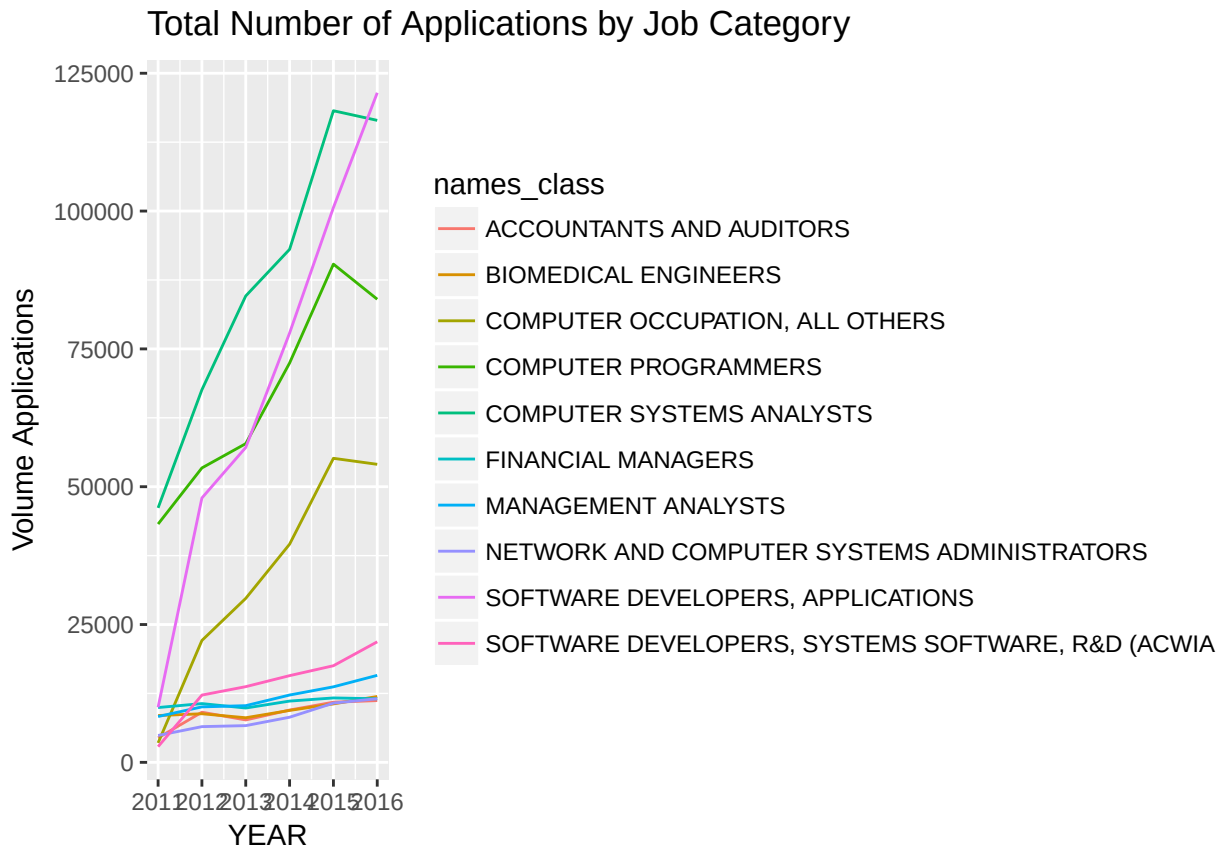
Investigations

Time Series Plots of the 10 Highest Volume Jobs

Looking at only the ten most frequently applied for job categories over all years we plot volume by year. Following that plot, we repeat the process but only count certified applications.

```
#using aggregate functions plot the top ten most frequently applied for visa job
#categories (SOC Name)
h1b_agg<-aggregate(SOC_NAME~YEAR+names_class, h1b_kaggle, FUN=length, na.action = NULL)
h1b_agg2<-aggregate(SOC_NAME~names_class, h1b_agg, FUN=sum)
h1b_agg2<-h1b_agg2[order(-h1b_agg2$SOC_NAME),]
h1b_aggfinal<-h1b_agg[h1b_agg$names_class %in% h1b_agg2$names_class[1:10], ]
h1b_aggnames<-cbind(unique(h1b_aggfinal$names_class), h1b_kaggle[match(unique(h1b_aggfinal$names_class)
h1b_aggfinal$names_class<-h1b_aggnames[match(h1b_aggfinal$names_class, h1b_aggnames[,1]),2]

#ggplot argument
ggplot(h1b_aggfinal, aes(x=YEAR, y=SOC_NAME, colour=names_class))+geom_line()+ylab("Volume Applications
```

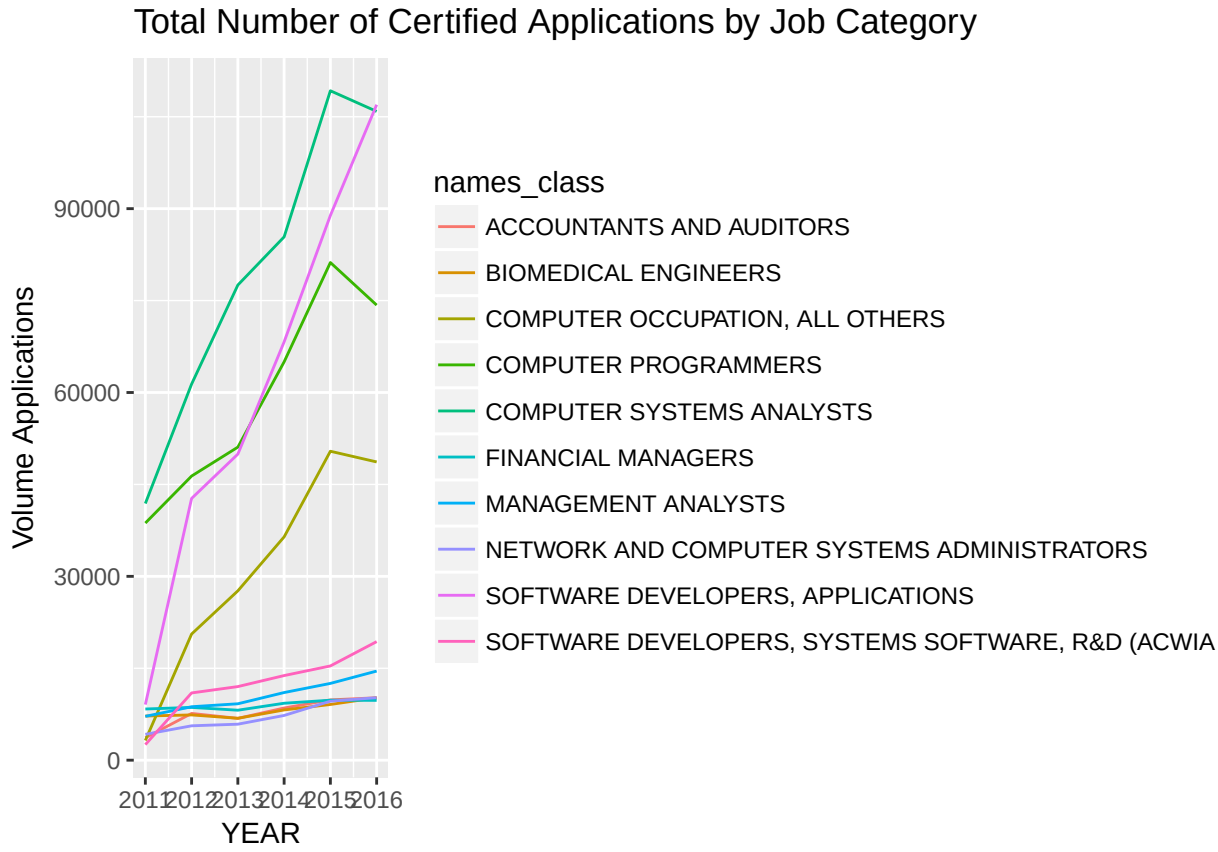



```
################

#same thing but now only counting certified cases (but using the same top ten categories)
h1bcert_agg<-aggregate(SOC_NAME~YEAR+names_class, h1b_kaggle[h1b_kaggle$CASE_STATUS=="CERTIFIED",], FUN=
h1bcert_agg2<-aggregate(SOC_NAME~names_class, h1bcert_agg, FUN=sum)
h1bcert_agg2<-h1bcert_agg2[order(-h1bcert_agg2$SOC_NAME),]
h1bcert_aggfinal<-h1bcert_agg[h1bcert_agg$names_class %in% h1bcert_agg2$names_class[1:10], ]

#technically redundant since were using the same ordered list of job categories as before
#h1b_aggnames<-cbind(unique(h1b_aggfinal$names_class), h1b_kaggle[match(unique(h1b_aggfinal$names_class
h1bcert_aggfinal$names_class<-h1b_aggnames[match(h1bcert_aggfinal$names_class, h1b_aggnames[,1]),2]

#similar plot
ggplot(h1bcert_aggfinal, aes(x=YEAR, y=SOC_NAME, colour=names_class))+geom_line()+ylab("Volume Applicati
```



Interestingly enough the plots look identical except for scale. This leads us to believe that the proportion of certified applications for each of these jobs over each year was approximately constant. We investigate this question again for all job categories (rather than just the top 10 most frequent) and find this to be true generally.

Fastest Growing Occupations

We produce dataframes showing the percent change over the previous year in applications for each job category along with the total volume of applications for that year.
In order to do this we had to take the intersection of the job category lists over all years- only considering job categories that had applications for every year in this dataset.

```
#We need to clean up the h1b_agg dataframe - which counts the number of occurences of each names_class(
namesclass_intersection<-Reduce(intersect, list(h1b_agg[h1b_agg$YEAR==2011,"names_class"],h1b_agg[h1b_ag
h1b_agg<-h1b_agg[h1b_agg$names_class %in% namesclass_intersection,]

#Produce 5 different dataframes with the percent change for each names_class(job category)- these have
for (i in 2012:2016){
      assign(paste("PercentChange_",i,sep=""), data.frame("JobCategory"=h1b_kaggle[unique(match(h1b_agg
}
df12<-PercentChange_2012[PercentChange_2012$Total > 1000,]
df12[order(-df12$PercentChange),][1:10,]
```

```
##                                      JobCategory PercentChange Total
## 82                  COMPUTER OCCUPATION, ALL OTHERS     527.86140 22107
## 75                SOFTWARE DEVELOPERS, APPLICATIONS     380.03804 47951
```

```
## 85 INFORMATION SECURITY ANALYSTS, WEB DEVELOPERS, AND     361.86186  1538
## 77  SOFTWARE DEVELOPERS, SYSTEMS SOFTWARE, R&D (ACWIA     328.11513 12197
## 52 MARKET RESEARCH ANALYSTS AND MARKETING SPECIALISTS     295.99084  6914
## 80       COMPUTER AND INFORMATION RESEARCH SCIENTISTS     288.84758  1046
## 29             ARCHITECTURAL AND ENGINEERING MANAGERS     280.50847  1347
## 14                          ACCOUNTANTS AND AUDITORS     104.88959  9093
## 98                       COMPUTER HARDWARE ENGINEERS      99.29947  1138
## 47                         COMPUTER SYSTEMS ANALYSTS      46.42733 67585
```

```
df13<-PercentChange_2013[PercentChange_2013$Total > 1000,]
df13[order(-df13$PercentChange),][1:10,]
```

```
##                                       JobCategory PercentChange Total
## 82                     COMPUTER OCCUPATION, ALL OTHERS      34.54562 29744
## 85  INFORMATION SECURITY ANALYSTS, WEB DEVELOPERS, AND      28.86866  1982
## 47                           COMPUTER SYSTEMS ANALYSTS      25.18606 84607
## 100            ELECTRONICS ENGINEERS, EXCEPT COMPUTER      19.98833  6171
## 171        HEALTH SPECIALTIES TEACHERS, POSTSECONDARY      19.13580  1544
## 75                  SOFTWARE DEVELOPERS, APPLICATIONS      19.12160 57120
## 19                                LOGISTICS ANALYSTS      14.23077  1188
## 77   SOFTWARE DEVELOPERS, SYSTEMS SOFTWARE, R&D (ACWIA      12.56866 13730
## 1                                                <NA>      12.35955  3600
## 120         MEDICAL SCIENTISTS EXCEPT EPIDEMIOLOGISTS      10.57550  4496
```

```
df14<-PercentChange_2014[PercentChange_2014$Total > 1000,]
df14[order(-df14$PercentChange),][1:10,]
```

```
##                                   JobCategory PercentChange Total
## 83                              WEB DEVELOPERS     818.13725  3746
## 75          SOFTWARE DEVELOPERS, APPLICATIONS      36.33228 77873
## 66                   DATABASE ADMINISTRATORS      34.99444  7287
## 82            COMPUTER OCCUPATION, ALL OTHERS      33.03860 39571
## 90      ARCHITECTS, EXCEPT LANDSCAPE AND NAVAL      25.80645  1014
## 224        COMMERCIAL AND INDUSTRIAL DESIGNERS      25.77049  1918
## 70                       COMPUTER PROGRAMMERS      25.30012 72436
## 9                              SALES MANGERS      23.65738  1819
## 14                    ACCOUNTANTS AND AUDITORS      23.33897  9486
## 78  NETWORK AND COMPUTER SYSTEMS ADMINISTRATORS      22.83441  8182
```

```
df15<-PercentChange_2015[PercentChange_2015$Total > 1000,]
df15[order(-df15$PercentChange),][1:10,]
```

```
##                                   JobCategory PercentChange  Total
## 83                              WEB DEVELOPERS      41.88468   5315
## 82            COMPUTER OCCUPATION, ALL OTHERS      39.36721  55149
## 64           FINANCIAL SPECIALISTS, ALL OTHER      38.63405   1360
## 78  NETWORK AND COMPUTER SYSTEMS ADMINISTRATORS      31.31264  10744
## 75          SOFTWARE DEVELOPERS, APPLICATIONS      29.23735 100641
## 87               OPERATIONS RESEARCH ANALYSTS      27.77178   7099
## 47                 COMPUTER SYSTEMS ANALYSTS      27.02136 118201
## 44                          MATHEMATICIANS      25.19648   5257
## 70                       COMPUTER PROGRAMMERS      24.76945  90378
## 255                       PHYSICAL THERAPISTS      22.19847   4002
```

```
df16<-PercentChange_2016[PercentChange_2016$Total > 1000,]
df16[order(-df16$PercentChange),][1:10,]
```

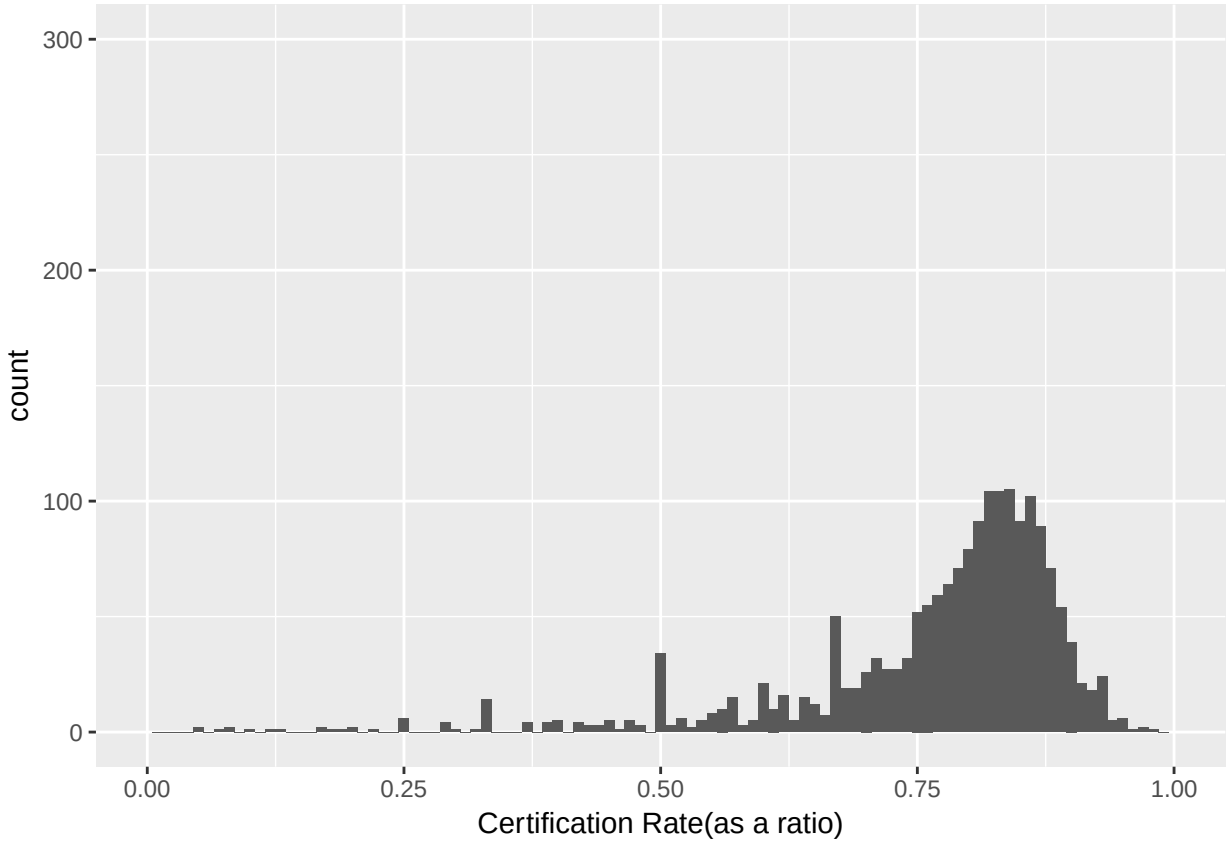
```
##                                        JobCategory PercentChange  Total
## 68              COMPUTER SYSTEMS ENGINEERS/ARCHITECTS     493.02326   1785
## 67   SOFTWARE QUALITY ASSURANCE ENGINEERS AND TESTERS     277.04026   3465
## 77  SOFTWARE DEVELOPERS, SYSTEMS SOFTWARE, R&D (ACWIA      24.75180  21864
## 64                    FINANCIAL SPECIALISTS, ALL OTHER      21.17647   1648
## 75                   SOFTWARE DEVELOPERS, APPLICATIONS      20.68739 121461
## 46                                MANAGEMENT ANALYSTS      15.18359  15779
## 12            COMPUTER & INFORMATION SYSTEMS MANAGERS      14.58410   6199
## 243     MEDICAL AND CLINICAL LABORATORY TECHNOLOGISTS      13.82743   1029
## 171        HEALTH SPECIALTIES TEACHERS, POSTSECONDARY      13.46267   1930
## 87                         OPERATIONS RESEARCH ANALYSTS      13.07226   8027
```

Not surprisingly, almost all of the fastest growing job categories are Computer Science/Technology occupations. As a technical note, we disregarded job categories with less thatn 1000 applications for any given year in order to avoid anomalous growth rates.

Histogram of Certification rate over all Job Categories

We aggregate the numbers of certified cases over the total number of cases for every job category (defined by the names_class varable) to look for anomalies in certification rates. The distribution turns out to be skewed normal indicating that different job categories are not considered preferentially for certification.

```
#We need to clean up the h1b_agg dataframe - which counts the number of occurences of each names_class(
namesclass_intersection<-Reduce(intersect, list(h1b_agg[h1b_agg$YEAR==2011,"names_class"],h1b_agg[h1b_ag
h1b_agg3<-h1b_agg[h1b_agg$names_class %in% namesclass_intersection,]
namesclass_intersection2<-Reduce(intersect, list(h1bcert_agg[h1bcert_agg$YEAR==2011,"names_class"],h1bc
h1bcert_agg3<-h1bcert_agg[h1bcert_agg$names_class %in% namesclass_intersection2,]
#h1bcert_agg3 has 108 fewer rows than h1b_agg3, indicating that some rows were excluded due to certain
#we want to add in these zero values. To do so, we have to know the names_class values of the excluded
namesclass_diff <- setdiff(unique(h1b_agg3$names_class),unique(h1bcert_agg3$names_class))
#now, we want to add 108 rows with names_class values equal to these to the h1bcert_agg3 df
cert_zeroes_mat <- cbind(rep(2011:2016,18),rep(namesclass_diff,each=6),rep(0,108))
colnames(cert_zeroes_mat) = c("YEAR","names_class","SOC_NAME")
zeroes_df <- as.data.frame(cert_zeroes_mat)
h1bcert_zeroes <- rbind(h1bcert_agg3,zeroes_df)
h1bcert_zeroes <- h1bcert_zeroes[order(h1bcert_zeroes$names_class),]
h1b_agg3 <- h1b_agg3[order(h1b_agg3$names_class),]
qplot(h1bcert_zeroes$SOC_NAME/h1b_agg3$SOC_NAME,geom="histogram",binwidth=1/100,xlim=c(0,1.0),ylim=c(0,
```


300
200
100
0
count
0.00
0.25
0.50
0.75
1.00
Certification Rate(as a ratio)