Project 1 - Data Syndicate Assignment / Jessie Alwerdt

Jessie Alwerdt

7/19/2020

#Data is from https://www.kaggle.com/paultimothymooney/phd-stipends/data# (https://www.kaggle.com/paultimothymooney/phd-stipends/data#) belonging to Paul Mooney

```
getwd()
```

```
## [1] "C:/Users/alwer/Documents/Data Syndicate Projects DSS - FB group/Code"
```

```
setwd("C:/Users/alwer/Documents/Data Syndicate Projects DSS - FB group/Data")
```

#Import dataset

```
library("readxl")
csv <- read_excel("C:/Users/alwer/Documents/Data Syndicate Projects DSS - FB group/Data/CSV_Excel.xls
x")
```

#Check to see what data type each variable is

```
str(csv)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    8707 obs. of  12 variables:
##  $ ID                : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ University        : chr  "University of Tennessee - Knoxville" "University of Arizona (U of A)"
"University of Arizona (U of A)" "University of Florida" ...
##  $ Department        : chr  "Economics" "Economics" "Economics" "ECE" ...
##  $ Overall_Pay       : num  1850 21000 19800 22600 26000 ...
##  $ LW_Ratio          : num  0.09 1.01 0.96 NA 1.14 1.2 1.69 0.87 1.1 1.13 ...
##  $ Academic_Year     : chr  "2020-2021" "2020-2021" "2020-2021" "2020-2021" ...
##  $ Program_Year      : chr  "1st" "1st" "1st" "1st" ...
##  $ Twelve_M_Gross_Pay: num  NA NA NA 24000 26000 ...
##  $ Nine_M_Gross_Pay  : num  1850 21000 21000 NA NA ...
##  $ 3_M_Gross_Pay     : num  NA NA NA NA NA NA NA NA NA 3500 ...
##  $ Fees              : num  NA NA 1200 1400 NA NA NA NA NA NA ...
##  $ Comments          : chr  NA "$10500 per semester. RA/TA required. Need to pay ~$600 per semeste
r." NA NA ...
```

```
csv1 <- csv
```

```
csv1$Academic_Year <- as.factor(csv1$Academic_Year)
is.factor(csv$Academic_Year)
```

```
## [1] FALSE
```

```
levels(csv1$Academic_Year)
```

```
##  [1] "2002-2003" "2003-2004" "2004-2005" "2005-2006" "2006-2007"
##  [6] "2007-2008" "2008-2009" "2009-2010" "2010-2011" "2011-2012"
## [11] "2012-2013" "2013-2014" "2014-2015" "2015-2016" "2016-2017"
## [16] "2017-2018" "2018-2019" "2019-2020" "2020-2021" "2021-2022"
```

```
csv1$Program_Year <- as.factor(csv1$Program_Year)
is.factor(csv$Program_Year)
```

```
## [1] FALSE
```

```
levels(csv1$Program_Year)
```

```
## [1] "1st"        "2nd"        "3rd"        "4th"        "5th"
## [6] "6th and up"
```

#8707 Cases

#Missing #Overall Pay: 21 missing / Min. -900000 / Max. 994000 / Mean 25124 #LW_Ratio: 911 Missing / Min. -34.01 / Max. 40.97 / Mean 1.076 #Academic Year: 4 missing #Program Year: 1020 missing #12 M Gross Pay: 2498 missing / Min. 1 / Max. 100000 / Mean = 28240 #9 M Gross Pay: 6233 missing / Min. 5 / Max. 189600 / Mean = 19596 #3 M Gross Pay: 7909 missing / Min. 3 / Max 55816 / Mean = 5043 #Fees: 5355 / Min. 1 / Max. 1000000 / Mean = 2870

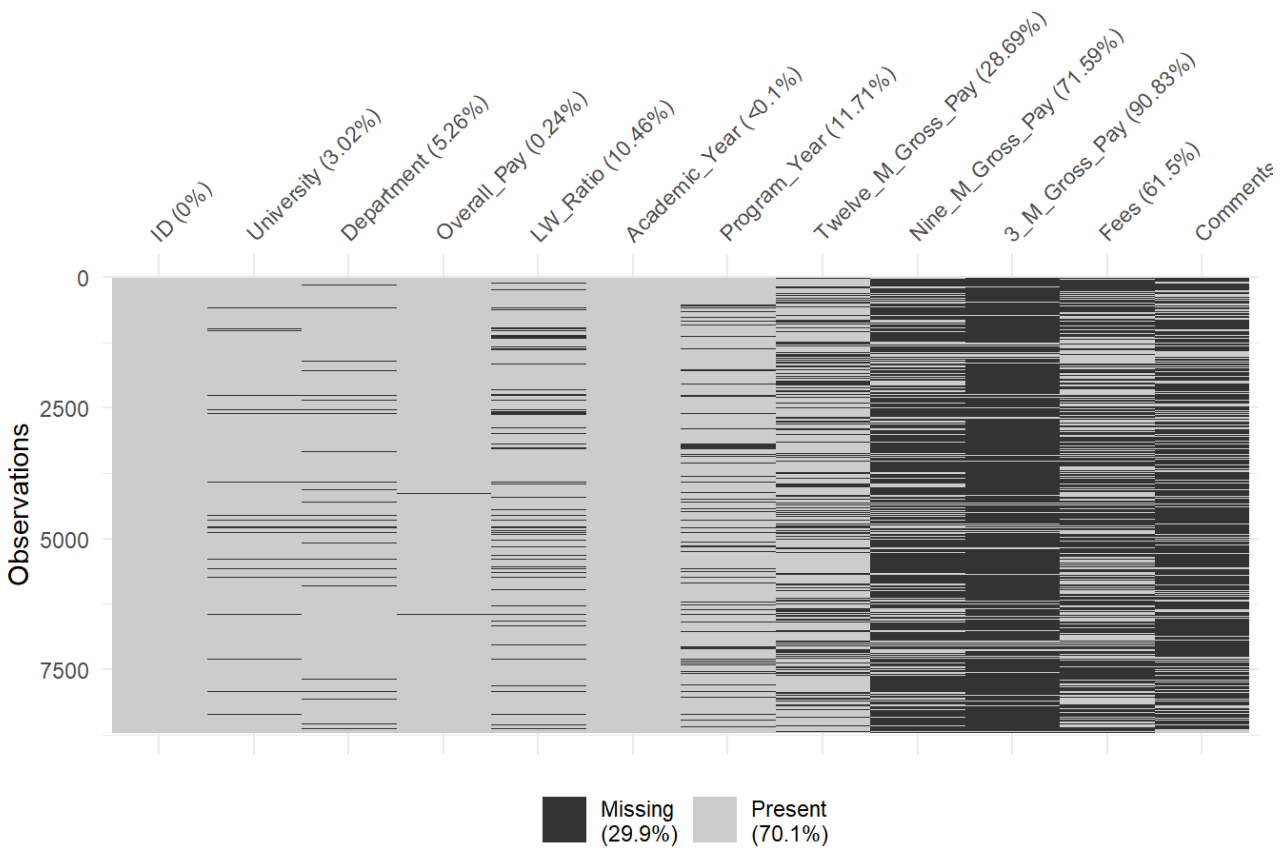
```
summary(csv1)
```

```
##        ID        University         Department         Overall_Pay
##  Min.   :   1   Length:8707        Length:8707        Min.   :-900000
##  1st Qu.:2178   Class :character   Class :character   1st Qu.:  19036
##  Median :4354   Mode  :character   Mode  :character   Median :  25000
##  Mean   :4354                                         Mean   :  25124
##  3rd Qu.:6530                                         3rd Qu.:  30400
##  Max.   :8707                                         Max.   : 994000
##                                                       NA's   :21
##     LW_Ratio         Academic_Year      Program_Year  Twelve_M_Gross_Pay
##  Min.   :-34.010   2016-2017:2198   1st       :4368   Min.   :      1
##  1st Qu.:  0.850   2018-2019:1911   2nd       :1071   1st Qu.:  23000
##  Median :  1.100   2019-2020:1369   3rd       : 841   Median :  28000
##  Mean   :  1.076   2017-2018:1197   4th       : 660   Mean   :  28240
##  3rd Qu.:  1.300   2020-2021: 767   5th       : 504   3rd Qu.:  32000
##  Max.   : 40.970   (Other)  :1261   6th and up: 243   Max.   :1000000
##  NA's   :911       NA's     :   4   NA's      :1020   NA's   :2498
##  Nine_M_Gross_Pay 3_M_Gross_Pay        Fees           Comments
##  Min.   :     5   Min.   :    3   Min.   :      1   Length:8707
##  1st Qu.: 15900   1st Qu.: 3000   1st Qu.:    500   Class :character
##  Median : 19000   Median : 4500   Median :   1006   Mode  :character
##  Mean   : 19596   Mean   : 5043   Mean   :   2870
##  3rd Qu.: 23000   3rd Qu.: 6000   3rd Qu.:   2000
##  Max.   :189600   Max.   :55816   Max.   :1000000
##  NA's   :6233     NA's   :7909    NA's   :5355
```

```
library(naniar)
```

```
## Warning: package 'naniar' was built under R version 3.6.3
```

```
vis_miss(csv1)
```



#Detect outliers

```
csv2 <- as.matrix(csv1)
```

```
library(devtools)
```

```
## Loading required package: usethis
```

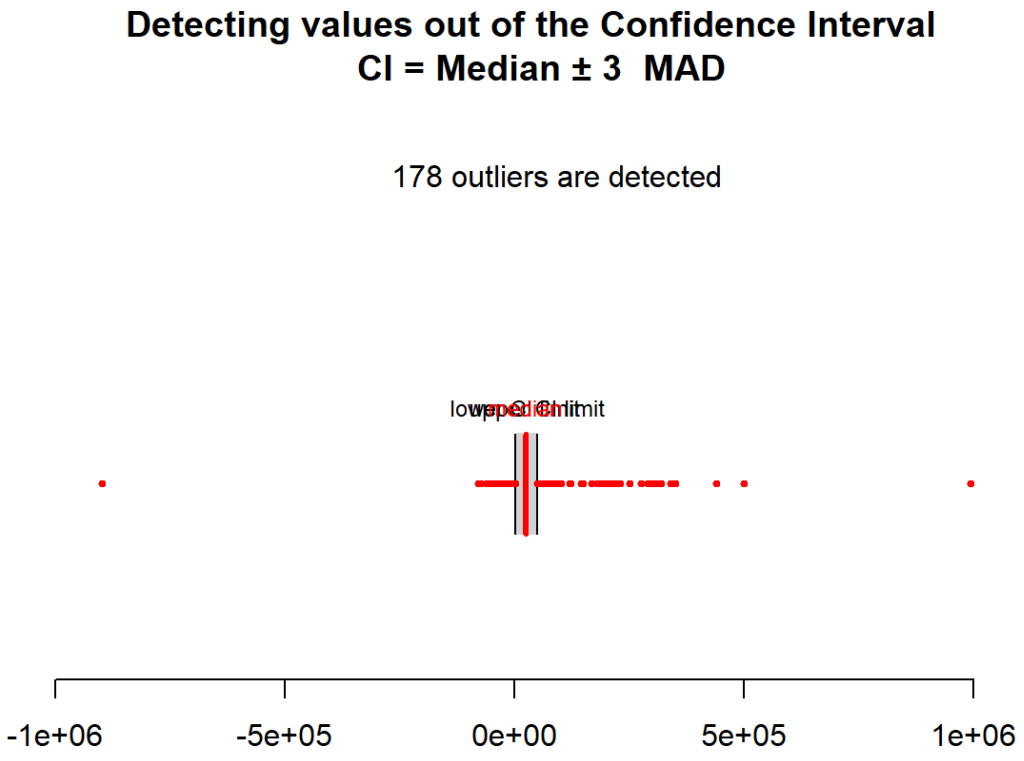
```
#install_github("mdelacre/Routliers")
```

#Outliers for overall pay #Total of 178 outliers

```
library(Routliers)
res1 <- outliers_mad(x = csv1$Overall_Pay)
res1
```

```
## Call:
## outliers_mad.default(x = csv1$Overall_Pay)
##
## Median:
## [1] 25000
##
## MAD:
## [1] 8154.3
##
## Limits of acceptable range of values:
## [1]   537.1 49462.9
##
## Number of detected outliers
##  extremely low extremely high          total
##             95             83            178
```

```
plot_outliers_mad(res1, x = csv1$Overall_Pay)
```

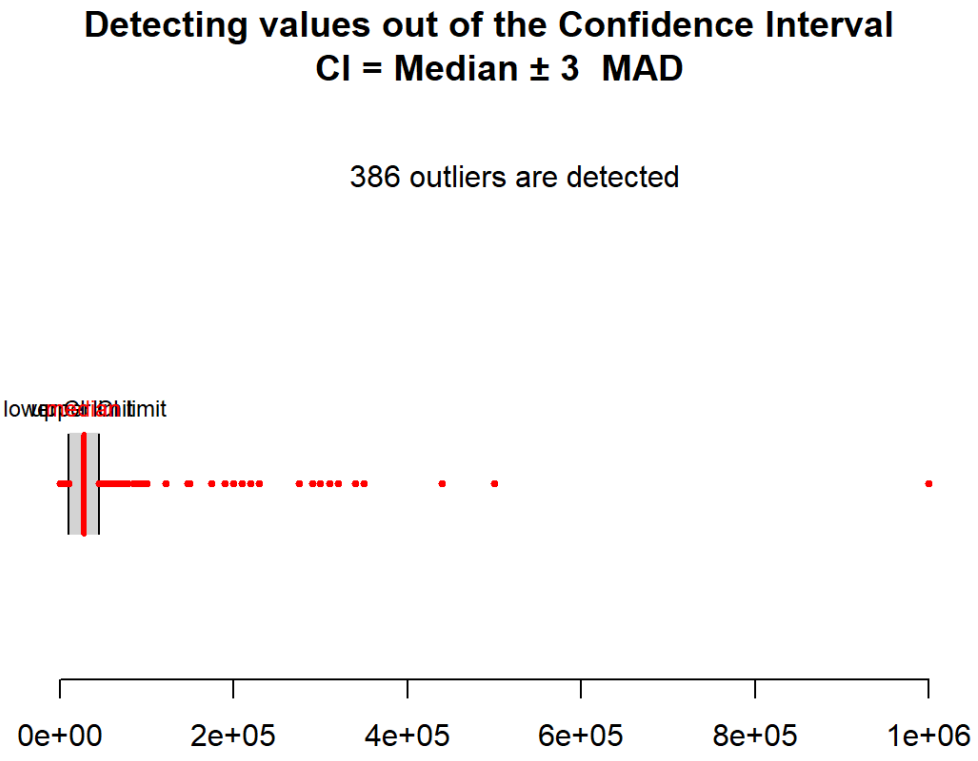


#Outliers for 12_M_Gross_Pay #Total of 386 Outliers

```
library(Routliers)
res1 <- outliers_mad(x = csv1$Twelve_M_Gross_Pay)
res1
```

```
## Call:
## outliers_mad.default(x = csv1$Twelve_M_Gross_Pay)
##
## Median:
## [1] 28000
##
## MAD:
## [1] 5930.4
##
## Limits of acceptable range of values:
## [1] 10208.8 45791.2
##
## Number of detected outliers
##  extremely low extremely high          total
##            261            125            386
```

```
plot_outliers_mad(res1, x = csv1$Twelve_M_Gross_Pay)
```

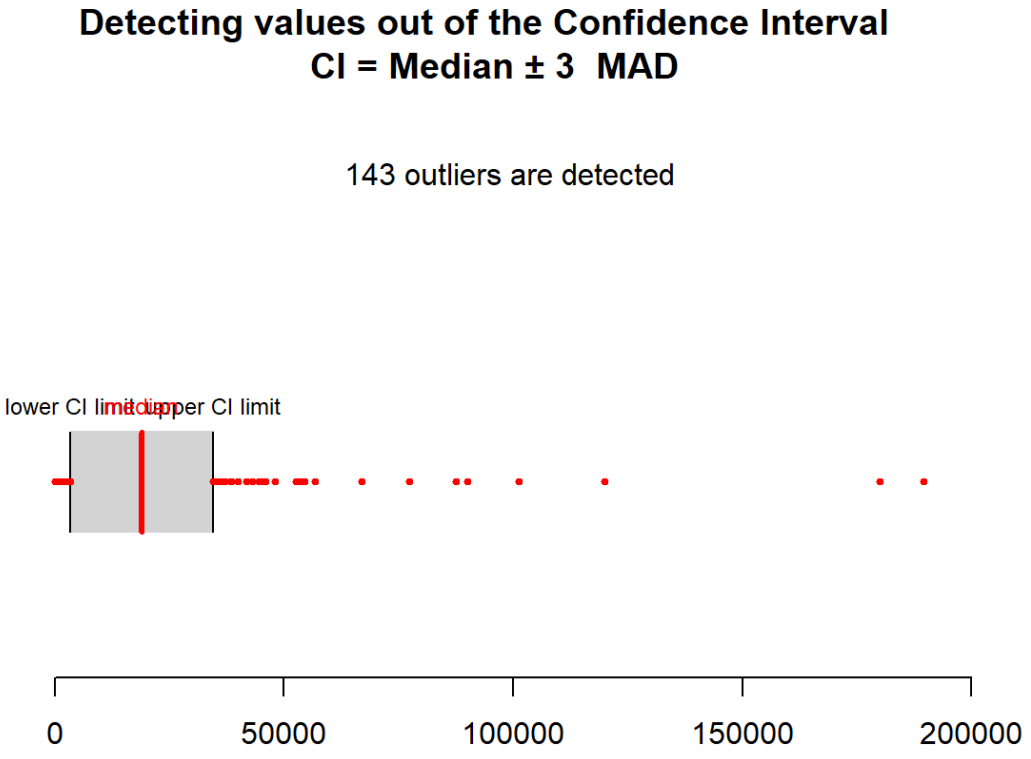


#Outliers for 9_M_Gross_Pay #Total of 143 Outliers

```
library(Routliers)
res1 <- outliers_mad(x = csv1$Nine_M_Gross_Pay)
res1
```

```
## Call:
## outliers_mad.default(x = csv1$Nine_M_Gross_Pay)
##
## Median:
## [1] 19000
##
## MAD:
## [1] 5189.1
##
## Limits of acceptable range of values:
## [1]  3432.7 34567.3
##
## Number of detected outliers
##  extremely low extremely high          total
##             96             47            143
```

```
plot_outliers_mad(res1, x = csv1$Nine_M_Gross_Pay)
```



#Had to rename variable due to not be able to reference it beginning with a number

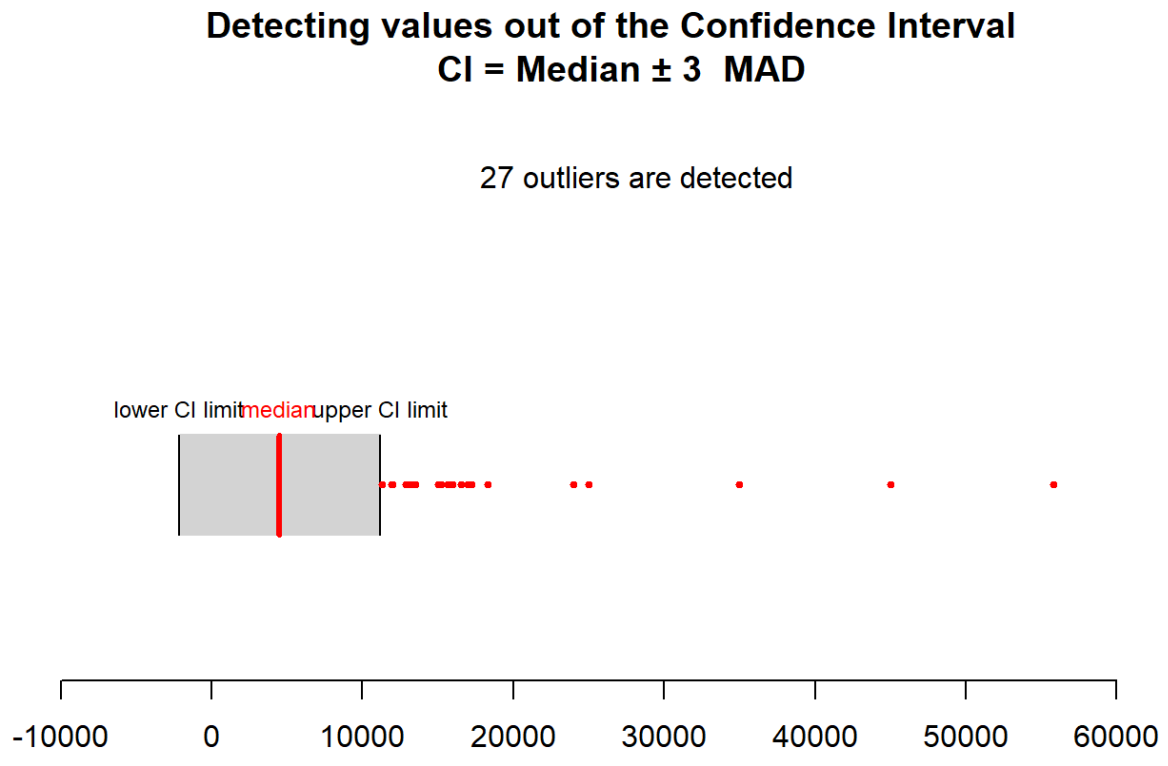
```
library(plyr)
csv1 <- rename(csv1, c("3_M_Gross_Pay" = "Three_M_Gross_Pay"))
```

#Outliers for 3_M_Gross_Pay #Total of 27 Outliers

```
library(Routliers)
res1 <- outliers_mad(x = csv1$Three_M_Gross_Pay)
res1
```

```
## Call:
## outliers_mad.default(x = csv1$Three_M_Gross_Pay)
##
## Median:
## [1] 4500
##
## MAD:
## [1] 2223.9
##
## Limits of acceptable range of values:
## [1] -2171.7 11171.7
##
## Number of detected outliers
##  extremely low extremely high          total
##              0             27             27
```

```
plot_outliers_mad(res1, x = csv1$Three_M_Gross_Pay)
```




```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.6.2
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
table(csv1$Academic_Year, csv1$Program_Year)
```

```
##
##              1st  2nd  3rd  4th  5th 6th and up
##   2002-2003    2    1    1    1    0          0
##   2003-2004    2    1    0    1    0          0
##   2004-2005    2    1    0    0    0          0
##   2005-2006    1    2    1    0    0          0
##   2006-2007    8    1    2    2    0          1
##   2007-2008    9    1    0    2    1          3
##   2008-2009    3    1    1    0    2          1
##   2009-2010    5    3    0    2    2          0
##   2010-2011    6    3    4    3    3          4
##   2011-2012   12    5    6    7    8          0
##   2012-2013   10    5    4    3   13          5
##   2013-2014   20    8   13   16   18         18
##   2014-2015  151   89   86   77   67         33
##   2015-2016  211   53   45   43   31         12
##   2016-2017 1094  248  182  132  108         49
##   2017-2018  776  117   85   47   61         26
##   2018-2019  710  332  255  194  121         58
##   2019-2020  833  150  119   93   45         24
##   2020-2021  511   50   37   37   24          8
##   2021-2022    2    0    0    0    0          0
```

#Highest year was 2016-2017 (2016 - 2020)

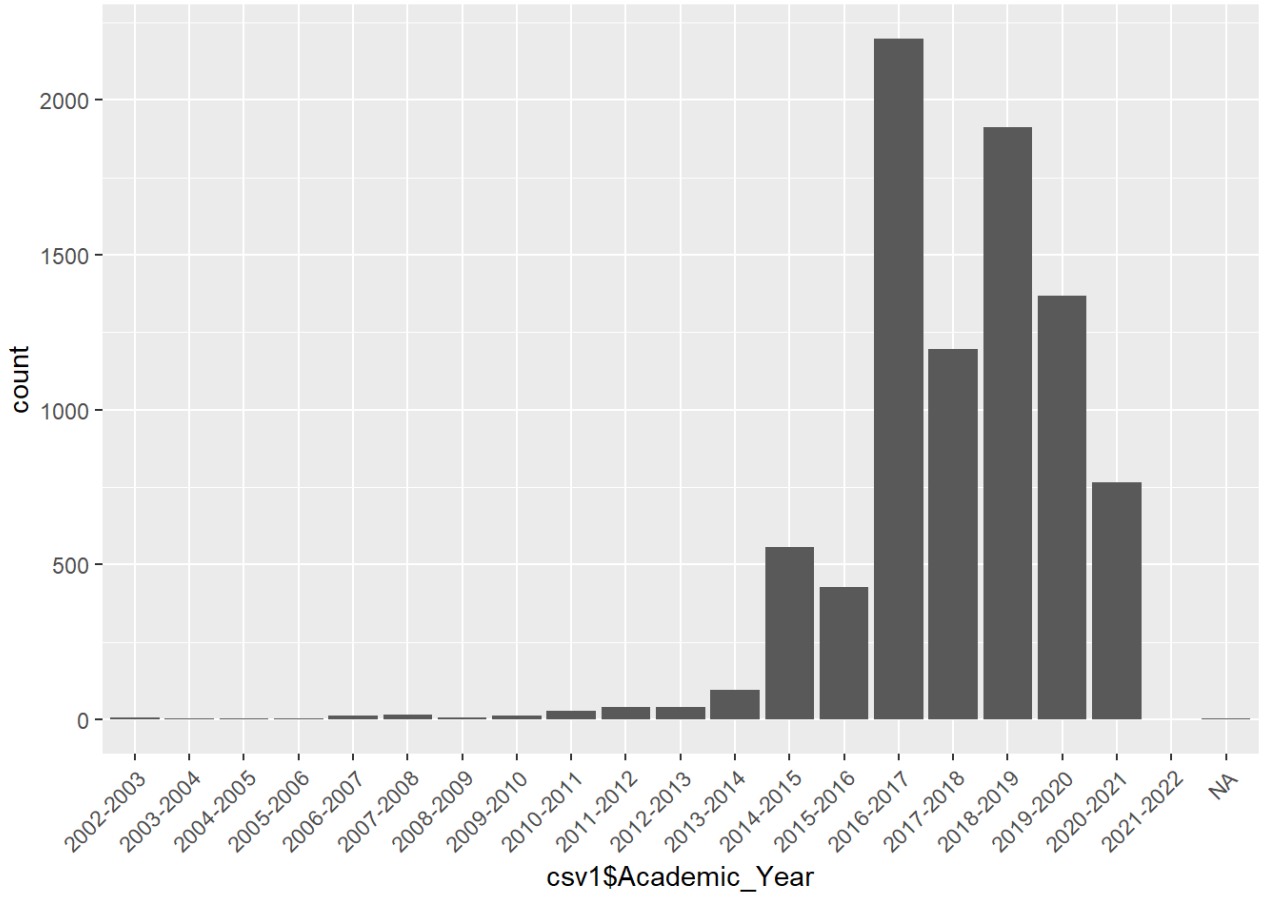
```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.2
```

```
table2 <- table(csv1$Academic_Year)
prop.table(table2)
```

```
##
##    2002-2003    2003-2004    2004-2005    2005-2006    2006-2007
## 0.0006894174 0.0004596116 0.0003447087 0.0005745145 0.0016086407
##    2007-2008    2008-2009    2009-2010    2010-2011    2011-2012
## 0.0018384465 0.0009192233 0.0016086407 0.0031023785 0.0047110192
##    2012-2013    2013-2014    2014-2015    2015-2016    2016-2017
## 0.0047110192 0.0111455820 0.0638860163 0.0490635413 0.2525565897
##    2017-2018    2018-2019    2019-2020    2020-2021    2021-2022
## 0.1375387797 0.2195794554 0.1573020797 0.0881305297 0.0002298058
```

```
ggplot(csv1, aes(x = csv1$Academic_Year)) +
        geom_bar() +
        theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

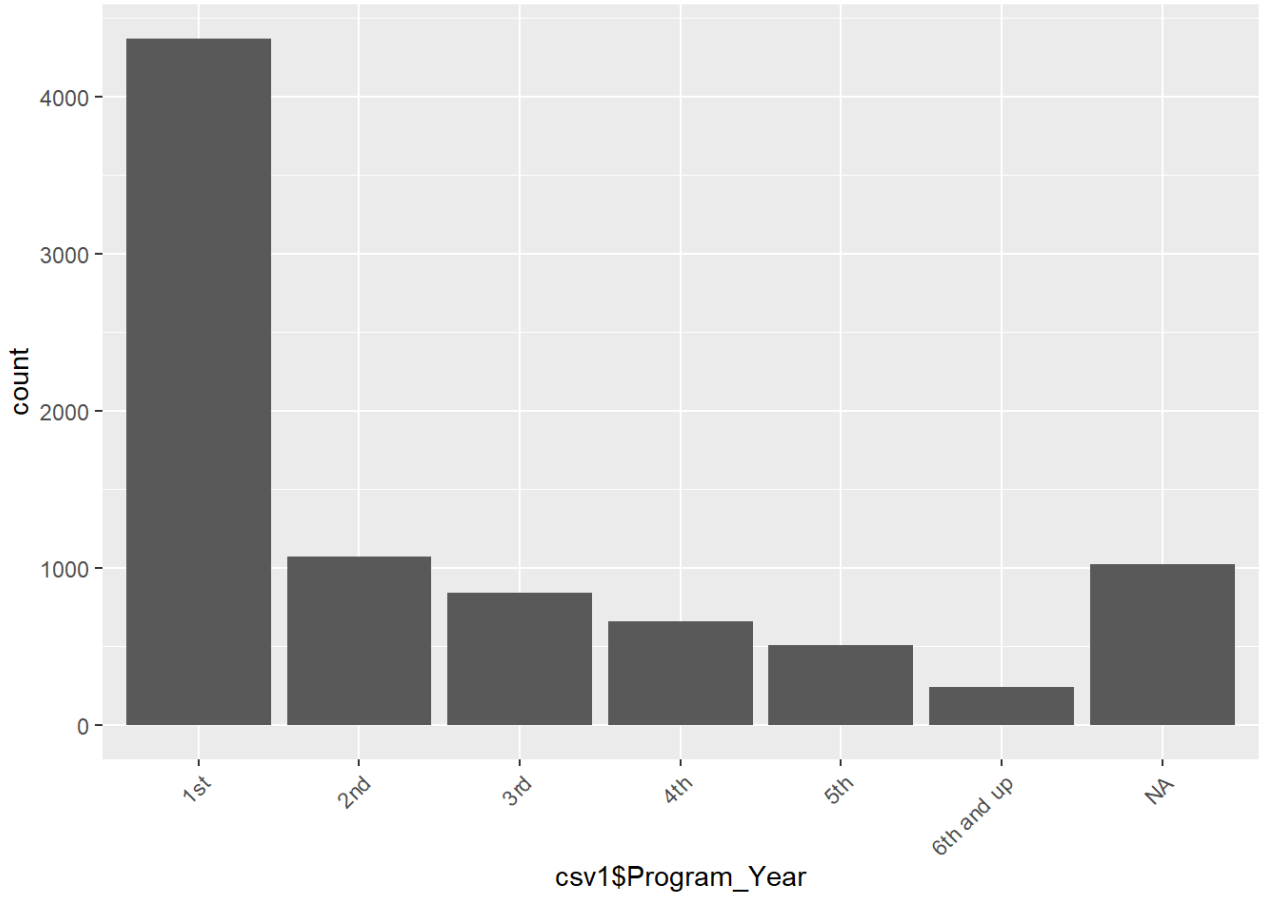


#Drop of cases with each program year

```
table2 <- table(csv1$Program_Year)
prop.table(table2)
```

```
##
##        1st        2nd        3rd        4th        5th 6th and up
## 0.56823208 0.13932614 0.10940549 0.08585924 0.06556524 0.03161181
```

```
ggplot(csv1, aes(x = csv1$Program_Year)) +
        geom_bar() +
        theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



#Look into the character values

```
library("wordcloud")
```

```
## Warning: package 'wordcloud' was built under R version 3.6.3
```

```
## Loading required package: RColorBrewer
```

```
library("tm")
```

```
## Warning: package 'tm' was built under R version 3.6.3
```

```
## Loading required package: NLP
```

```
##
## Attaching package: 'NLP'
```

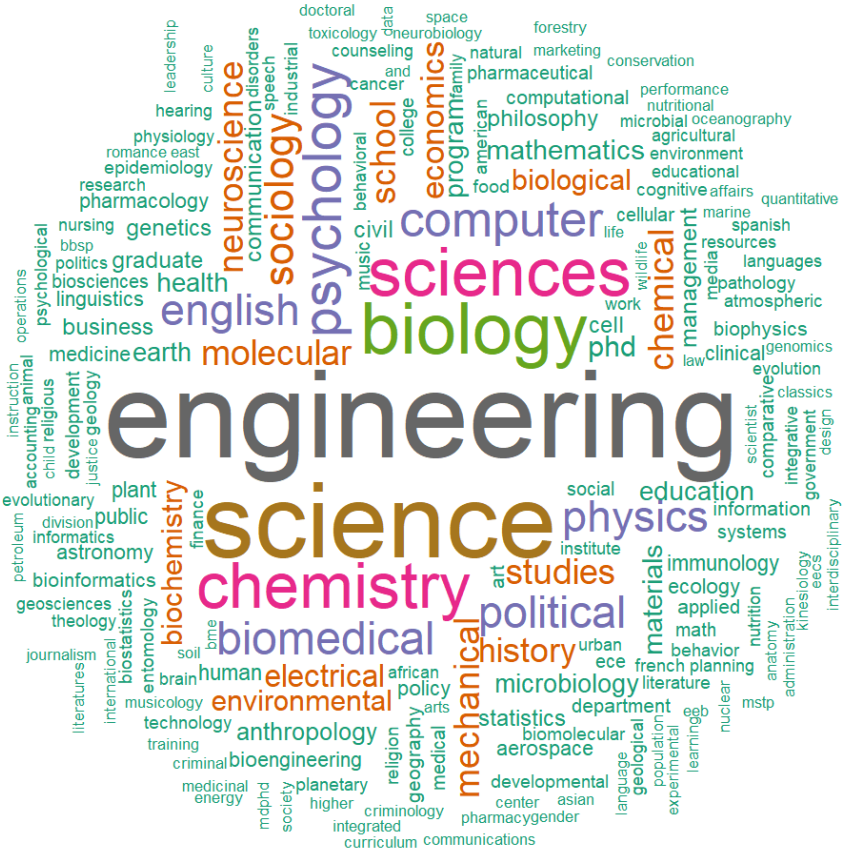
```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
wordcloud(words = csv1$University, min.freq = 1,
          max.words=200, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation):
## transformation drops documents
```

```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```


```
wordcloud(words = csv1$Department, min.freq = 1,
          max.words=200, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))
```



#Wanted to do so much more #Would like to replicate in SAS and Python environment #Would like to look at cleaning this better #Would like to look into the missing cases more #Outliers need to be examined closer. Some are most likely errors.