Data Science University MLM1805 Final Project

Jane Hung

August 17, 2018

CoNVO: Context, Need, Value, Outcome

Although we have multiple ways to understand and group our providers (i.e. through line of business, size, etc.), we still do not have a full understanding of how to segment our providers. In an environment where our providers feel like we do not know who they are and how to best serve their needs, it is imperative to gain more intuition into the true provider personas. As a healthcare insurance company, creating these provider segments allows us to understand a provider group's common concerns, background information, etc. and grants us the ability to interact with individual providers with unique insights into their issues. This analysis directly adds value to the company because we are then able to positively influence Net Promoter Scores (NPS) and develop a trusting relationship between us and our provider partners. As an outcome, this analysis could be used in call centers so our agents can tailor their conversation to the provider persona. Furthermore, this analysis could be used within our Advocate Network to help facilitate stronger communication and empathy.

Question

What provider personas can be formed through unsupervised clustering methods?

Hypothesis

Based on the data and completing exploratory data analysis, below are expected features that may determine provider clusters.
* Providers accepting Medicare assignments
* Providers using electronic health records
* Providers with a secondary specialty
* Providers with a hospital affiliation
* Providers in a hospital network

Data Lineage

For this analysis, open source provider data was used from the Centers for Medicare & Medicaid Services. Please vist here for the most up-to-date source. This data was used because it has robust documentation, available SMEs, and current data refreshes. Furthermore, the data was fairly clean and was in a format that was easy to complete feature engineering. The total file contains 2.67 million rows; however, due to issues with high performance computing using R on a local machine, I opted to compute a simple random sample to get 5% of the total data, which amounted to ~133 thousand observations. This dataset contains demographic and Medicare quality program participation information for individual eligible professionals.

Cohort Definition

Within this analysis, I opted to include all providers taken from the SRS of 5% of the total CMS dataset because these were all active and eligible providers. Furthermore, every provider in this dataset was represented as a single observation, and only individual providers (not organizations) were represented as a row. In addition, one additional constraint to this dataset is that it was taken from the CMS website, which indicates that these providers have had some affiliation with Medicare and Medicaid. Therefore, providers not in our cohort are those that *do not* have any past affiliation with CMS.

Initialize environment

Import and cache data

```
# Eventually I would like to directly source the information from the CMS website and cache it
# setCacheDir(tempdir())
#
# simpleCache('provider.data', {provider <- read.socrata("https://data.medicare.gov/resource/c8qv-268j.csv")
})

provider <- read_csv('Physician_Compare_National_Downloadable_File.csv',
                     col_types = paste(rep('c',41), sep = '',collapse = ''))
```

Check data quality and data types

Sample data to make data easier to work with initially. Get 5% of total dataset.

```
provider <- sample_n(provider,
                     size = ceiling(.05*nrow(provider)),
                     replace = TRUE)
```

Make valid names for columns, i.e. remove spaces in column names

```
names(provider) <- make.names(names(provider))
```

Change to numerical variables

```
# provider$NPI <- as.factor(provider$NPI)

provider$years.after.grad <- as.year(Sys.Date()) - as.year(provider$Graduation.year)
provider$Number.of.Group.Practice.members <- as.numeric(provider$Number.of.Group.Practice.members)
```

Check the provider dataframe

```
str(provider)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    133061 obs. of  42 variables:
##  $ NPI                                                      : chr  "1730519752" "1528148947" "1
669788949" "1508113853" ...
##  $ PAC.ID                                                   : chr  "7416184403" "2264629666" "0
749463453" "3678715455" ...
##  $ Professional.Enrollment.ID                               : chr  "I20131219000824" "I201012100
00903" "I20110317000580" "I20130807000177" ...
##  $ Last.Name                                                : chr  "STONE" "PENMETSA" "GRAY" "P
LESSNER" ...
##  $ First.Name                                               : chr  "ROBYN" "SANTHI" "TERRANCE"
"MELISSA" ...
##  $ Middle.Name                                              : chr  NA NA "WAYNE" "R" ...
##  $ Suffix                                                   : chr  NA NA NA NA ...
##  $ Gender                                                   : chr  "F" "F" "M" "F" ...
##  $ Credential                                               : chr  NA NA NA NA ...
##  $ Medical.school.name                                      : chr  "OTHER" "OTHER" "OTHER" "NEW
YORK COLLEGE OF OSTEO MEDICINE OF NEW YORK INSTITUTE OF TECHNOLOGY" ...
##  $ Graduation.year                                          : chr  "2013" "1993" "2010" "2010"
...
##  $ Primary.specialty                                        : chr  "PHYSICAL THERAPY" "RHEUMATOL
OGY" "CERTIFIED REGISTERED NURSE ANESTHETIST" "HOSPITALIST" ...
##  $ Secondary.specialty.1                                    : chr  NA "INTERNAL MEDICINE" NA "IN
TERNAL MEDICINE" ...
##  $ Secondary.specialty.2                                    : chr  NA NA NA NA ...
##  $ Secondary.specialty.3                                    : chr  NA NA NA NA ...
##  $ Secondary.specialty.4                                    : chr  NA NA NA NA ...
##  $ All.secondary.specialties                                : chr  NA "INTERNAL MEDICINE" NA "IN
TERNAL MEDICINE" ...
##  $ Organization.legal.name                                  : chr  "MILTON CHIROPRACTIC AND REHA
BILITATION INC" "ROCKDALE BLACKHAWK LLC" "ST JOHN HOSPITAL AND MEDICAL CENTER" "UT PHYSICIANS" ...
##  $ Group.Practice.PAC.ID                                    : chr  "0941113708" "2769487115" "31
73424082" "8426960360" ...
##  $ Number.of.Group.Practice.members                         : num  41 86 272 NA 146 6 118 38 35
460 ...
##  $ Line.1.Street.Address                                    : chr  "111 WILLARD ST" "611 W HWY 6
101" "22101 MOROSS RD" "1333 MOURSUND ST" ...
##  $ Line.2.Street.Address                                    : chr  "SUITE GA" NA NA "MEMORIAL HE
RMANN TIRR" ...
##  $ Marker.of.address.line.2.suppression                     : chr  NA NA NA NA ...
##  $ City                                                     : chr  "QUINCY" "WACO" "DETROIT" "H
OUSTON" ...
##  $ State                                                    : chr  "MA" "TX" "MI" "TX" ...
##  $ Zip.Code                                                 : chr  "021691200" "767107545" "482
362148" "770303405" ...
##  $ Phone.Number                                             : chr  "6174714491" "2547554582" "31
33434000" "7137975929" ...
##  $ Hospital.affiliation.CCN.1                               : chr  NA "451357" "230165" "450068"
...
##  $ Hospital.affiliation.LBN.1                               : chr  NA "LITTLE RIVER HEALTHCARE"
"ST JOHN HOSPITAL AND MEDICAL CENTER" "MEMORIAL HERMANN TEXAS MEDICAL CENTER" ...
##  $ Hospital.affiliation.CCN.2                               : chr  NA "451385" "230216" NA ...
##  $ Hospital.affiliation.LBN.2                               : chr  NA "GOODALL WITCHER HOSPITAL"
"MCLAREN PORT HURON" NA ...
##  $ Hospital.affiliation.CCN.3                               : chr  NA NA NA NA ...
##  $ Hospital.affiliation.LBN.3                               : chr  NA NA NA NA ...
##  $ Hospital.affiliation.CCN.4                               : chr  NA NA NA NA ...
##  $ Hospital.affiliation.LBN.4                               : chr  NA NA NA NA ...
##  $ Hospital.affiliation.CCN.5                               : chr  NA NA NA NA ...
##  $ Hospital.affiliation.LBN.5                               : chr  NA NA NA NA ...
##  $ Professional.accepts.Medicare.Assignment                 : chr  "Y" "Y" "Y" "Y" ...
##  $ Reported.Quality.Measures                                : chr  "Y" "Y" NA "Y" ...
##  $ Used.electronic.health.records                           : chr  NA "Y" NA NA ...
##  $ Committed.to.heart.health.through.the.Million.Hearts..initiative.: chr  NA NA NA NA ...
##  $ years.after.grad                                         : num  5 25 8 8 29 35 22 23 30 19 ..
.
```

Number of observations in this dataset: **133061**
Number of features in this dataset: **42**
Convert multiple character columns to factors

```r
for (i in names(provider)) {
  if (class(provider[[i]]) == "character") {
    provider[[i]] <- factor(provider[[i]])
  }
}
```

Conduct exploratory data analysis

Give top 10 counts of all columns and summary of numerical data

```r
for (i in names(provider)) {
  cat("\n")
  if (class(provider[[i]]) == "numeric") {
    print(i)
    print(summary(provider[[i]]))
  } else {
    provider %>%
      group_by_(.dots = i) %>%
      dplyr::summarize(num_provider = n()) %>%
      arrange(desc(num_provider)) %>%
      head(n = 10) %>%
      print()
  }
}
```

```
##
## # A tibble: 10 x 2
##    NPI        num_provider
##    <fct>             <int>
##  1 1275731010           15
##  2 1578595815           10
##  3 1285608661            9
##  4 1104904655            8
##  5 1265408579            8
##  6 1760812796            8
##  7 1336359462            7
##  8 1720342355            7
##  9 1033256771            6
## 10 1063507713            6
##
## # A tibble: 10 x 2
##    PAC.ID     num_provider
##    <fct>             <int>
##  1 7416128319           15
##  2 1153301494           10
##  3 5294802104            9
##  4 1557599982            8
##  5 2567367121            8
##  6 6800883992            8
##  7 3870740269            7
##  8 5496849440            7
##  9 0244429884            6
## 10 0547332686            6
##
## # A tibble: 10 x 2
##    Professional.Enrollment.ID num_provider
##    <fct>                             <int>
##  1 I20110926000873                      15
##  2 I20040726000703                      10
##  3 I20090320000509                       9
##  4 I20031205000385                       8
##  5 I20070910000738                       8
##  6 I20140110000957                       8
##  7 I20120816000932                       7
##  8 I20120820000599                       7
##  9 I20031111000371                       6
## 10 I20040601000630                       6
##
## # A tibble: 10 x 2
##    Last.Name num_provider
```

```
##    <fct>            <int>
##  1 SMITH              661
##  2 PATEL              595
##  3 JOHNSON            556
##  4 LEE                497
##  5 MILLER             490
##  6 BROWN              363
##  7 WILLIAMS           355
##  8 JONES              334
##  9 KIM                314
## 10 ANDERSON           295
##
## # A tibble: 10 x 2
##    First.Name num_provider
##    <fct>             <int>
##  1 MICHAEL            2746
##  2 DAVID              2474
##  3 JOHN               2378
##  4 ROBERT             1936
##  5 JAMES              1744
##  6 WILLIAM            1364
##  7 MARK               1303
##  8 JENNIFER           1218
##  9 RICHARD            1205
## 10 THOMAS             1190
##
## # A tibble: 10 x 2
##    Middle.Name num_provider
##    <fct>              <int>
##  1 <NA>               32657
##  2 A                  10115
##  3 M                   9830
##  4 J                   7442
##  5 L                   7087
##  6 R                   5292
##  7 S                   4938
##  8 E                   4876
##  9 C                   4429
## 10 D                   4333
##
## # A tibble: 9 x 2
##   Suffix num_provider
##   <fct>         <int>
## 1 <NA>         130577
## 2 JR.            1520
## 3 III             551
## 4 II              257
## 5 IV               86
## 6 SR.              54
## 7 I                12
## 8 IX                2
## 9 V                 2
##
## # A tibble: 2 x 2
##   Gender num_provider
##   <fct>         <int>
## 1 M             75631
## 2 F             57430
##
## # A tibble: 10 x 2
##    Credential num_provider
##    <fct>             <int>
##  1 <NA>              88139
##  2 MD                31562
##  3 PA                 2766
##  4 NP                 2023
##  5 DO                 1866
##  6 CNA                1172
##  7 DC                 1030
##  8 PT                  995
##  9 OD                  955
## 10 CSW                 711
##
```

```
## # A tibble: 10 x 2
##    Medical.school.name                                     num_provider
##    <fct>                                                          <int>
##  1 OTHER                                                          65305
##  2 INDIANA UNIVERSITY SCHOOL OF MEDICINE                           1170
##  3 WAYNE STATE UNIVERSITY SCHOOL OF MEDICINE                       1135
##  4 UNIVERSITY OF MINNESOTA MEDICAL SCHOOL                           919
##  5 OHIO STATE UNIVERSITY COLLEGE OF MEDICINE                        864
##  6 UNIVERSITY OF ILLINOIS AT CHICAGO HEALTH SCIENCE CENTER          861
##  7 UNIVERSITY OF MICHIGAN MEDICAL SCHOOL                            853
##  8 TEMPLE UNIVERSITY SCHOOL OF MEDICINE                             849
##  9 PHILADELPHIA COLLEGE OF OSTEOPATHIC MEDICINE                     824
## 10 JEFFERSON MEDICAL COLLEGE OF THOMAS JEFFERSON UNIVERSITY         820
##
## # A tibble: 10 x 2
##    Graduation.year num_provider
##    <fct>                  <int>
##  1 2010                    4162
##  2 2009                    4138
##  3 2008                    4115
##  4 2007                    4100
##  5 2011                    3951
##  6 2002                    3924
##  7 2003                    3906
##  8 2001                    3903
##  9 2004                    3898
## 10 2000                    3850
##
## # A tibble: 10 x 2
##    Primary.specialty                    num_provider
##    <fct>                                       <int>
##  1 NURSE PRACTITIONER                          12674
##  2 INTERNAL MEDICINE                           11366
##  3 PHYSICIAN ASSISTANT                         11067
##  4 FAMILY PRACTICE                             10057
##  5 DIAGNOSTIC RADIOLOGY                         8560
##  6 PHYSICAL THERAPY                             5271
##  7 CERTIFIED REGISTERED NURSE ANESTHETIST       4308
##  8 CARDIOVASCULAR DISEASE (CARDIOLOGY)          4231
##  9 ANESTHESIOLOGY                               4124
## 10 OBSTETRICS/GYNECOLOGY                        3924
##
## # A tibble: 10 x 2
##    Secondary.specialty.1               num_provider
##    <fct>                                      <int>
##  1 <NA>                                      113925
##  2 INTERNAL MEDICINE                           7063
##  3 CARDIOVASCULAR DISEASE (CARDIOLOGY)         1385
##  4 CRITICAL CARE (INTENSIVISTS)                1264
##  5 PEDIATRIC MEDICINE                           788
##  6 GENERAL SURGERY                              620
##  7 FAMILY PRACTICE                              508
##  8 GERIATRIC MEDICINE                           424
##  9 INTERVENTIONAL RADIOLOGY                     414
## 10 EMERGENCY MEDICINE                           405
##
## # A tibble: 10 x 2
##    Secondary.specialty.2     num_provider
##    <fct>                            <int>
##  1 <NA>                            130957
##  2 INTERNAL MEDICINE                  949
##  3 PULMONARY DISEASE                  168
##  4 MEDICAL ONCOLOGY                    80
##  5 PAIN MANAGEMENT                     79
##  6 PEDIATRIC MEDICINE                  72
##  7 NUCLEAR MEDICINE                    65
##  8 VASCULAR SURGERY                    61
##  9 SLEEP MEDICINE                      59
## 10 INTERVENTIONAL CARDIOLOGY           42
##
## # A tibble: 10 x 2
##    Secondary.specialty.3       num_provider
##    <fct>                              <int>
```

```
##    <fct>                              <int>
##  1 <NA>                              132819
##  2 INTERNAL MEDICINE                     47
##  3 SLEEP MEDICINE                        34
##  4 PERIPHERAL VASCULAR DISEASE           19
##  5 NUCLEAR MEDICINE                      18
##  6 VASCULAR SURGERY                      13
##  7 PULMONARY DISEASE                     12
##  8 MEDICAL ONCOLOGY                      11
##  9 PEDIATRIC MEDICINE                    11
## 10 SPORTS MEDICINE                        9
##
## # A tibble: 10 x 2
##    Secondary.specialty.4          num_provider
##    <fct>                                 <int>
##  1 <NA>                                 133021
##  2 SLEEP MEDICINE                            5
##  3 INTERNAL MEDICINE                         4
##  4 VASCULAR SURGERY                          4
##  5 MEDICAL ONCOLOGY                          3
##  6 PAIN MANAGEMENT                           3
##  7 GENERAL PRACTICE                          2
##  8 INTERVENTIONAL PAIN MANAGEMENT            2
##  9 NEUROLOGY                                 2
## 10 OBSTETRICS/GYNECOLOGY                     2
##
## # A tibble: 10 x 2
##    All.secondary.specialties           num_provider
##    <fct>                                      <int>
##  1 <NA>                                      113925
##  2 INTERNAL MEDICINE                           6736
##  3 CARDIOVASCULAR DISEASE (CARDIOLOGY)         1009
##  4 PEDIATRIC MEDICINE                           787
##  5 CRITICAL CARE (INTENSIVISTS)                 771
##  6 GENERAL SURGERY                              581
##  7 FAMILY PRACTICE                              458
##  8 INTERVENTIONAL RADIOLOGY                     387
##  9 PAIN MANAGEMENT                              387
## 10 GERIATRIC MEDICINE                           363
##
## # A tibble: 10 x 2
##    Organization.legal.name                      num_provider
##    <fct>                                               <int>
##  1 <NA>                                                13303
##  2 REGENTS OF THE UNIVERSITY OF MICHIGAN                1322
##  3 THE CLEVELAND CLINIC FOUNDATION                      1012
##  4 SOUTHERN CALIFORNIA PERMANENTE MEDICAL GROUP          790
##  5 UNIVERSITY OF PITTSBURGH PHYSICIANS                   640
##  6 PERMANENTE MEDICAL GROUP INC                          630
##  7 IHC HEALTH SERVICES INC                               624
##  8 PHYSICIANS REFERRAL SERVICE                           574
##  9 NORTH SHORE - LIJ MEDICAL PC                          536
## 10 UNIVERSITY OF PENN MEDICAL GROUP                      501
##
## # A tibble: 10 x 2
##    Group.Practice.PAC.ID num_provider
##    <fct>                        <int>
##  1 <NA>                         13303
##  2 3779496856                    1322
##  3 1850203555                    1012
##  4 6002729175                     790
##  5 8729990239                     640
##  6 8921910225                     630
##  7 1850209420                     624
##  8 7911801410                     574
##  9 3375701568                     536
## 10 6204730955                     501
##
## [1] "Number.of.Group.Practice.members"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     2.0    16.0    78.0   197.6   288.0   995.0   34259
##
## # A tibble: 10 x 2
```

```
##    Line.1.Street.Address            num_provider
##    <fct>                                   <int>
##  1 10685 CARNEGIE AVE                        352
##  2 600 N WOLFE ST                            189
##  3 1515 HOLCOMBE BLVD CLINICAL LAB           180
##  4 200 1ST ST SW                             175
##  5 1515 HOLCOMBE BLVD HEMOPATHOLOGY          172
##  6 9500 EUCLID AVE                           167
##  7 111 E 210TH ST                            164
##  8 13123 E 16TH AVE                          159
##  9 9300 EUCLID AVE                           148
## 10 55 FRUIT ST                               147
##
## # A tibble: 10 x 2
##    Line.2.Street.Address num_provider
##    <fct>                        <int>
##  1 <NA>                         99255
##  2 SUITE 100                     1613
##  3 SUITE 200                     1401
##  4 SUITE 101                      863
##  5 SUITE 300                      629
##  6 SUITE 1                        535
##  7 SUITE 201                      501
##  8 SUITE 102                      492
##  9 SUITE A                        462
## 10 SUITE 2                        393
##
## # A tibble: 2 x 2
##   Marker.of.address.line.2.suppression num_provider
##   <fct>                                       <int>
## 1 <NA>                                       127520
## 2 Y                                            5541
##
## # A tibble: 10 x 2
##    City         num_provider
##    <fct>               <int>
##  1 HOUSTON              2085
##  2 NEW YORK             1678
##  3 ANN ARBOR            1323
##  4 PHILADELPHIA         1213
##  5 CHICAGO              1206
##  6 PITTSBURGH           1100
##  7 LOS ANGELES          1049
##  8 CLEVELAND             946
##  9 BALTIMORE             932
## 10 DALLAS                847
##
## # A tibble: 10 x 2
##    State num_provider
##    <fct>        <int>
##  1 CA           10754
##  2 TX            9452
##  3 NY            8532
##  4 PA            7891
##  5 FL            7452
##  6 MI            5805
##  7 IL            5507
##  8 OH            5470
##  9 NC            4186
## 10 MA            3760
##
## # A tibble: 10 x 2
##    Zip.Code  num_provider
##    <fct>            <int>
##  1 481095000         1128
##  2 441950001          667
##  3 770304000          452
##  4 326103003          311
##  5 110303816          270
##  6 152124756          238
##  7 900950001          230
##  8 212870005          187
##  9 559050001          175
```

```
## 10 104672401          164
##
## # A tibble: 10 x 2
##    Phone.Number num_provider
##    <fct>               <int>
##  1 <NA>                18642
##  2 8663892727            404
##  3 2164443475            352
##  4 7136204000            211
##  5 4154761000            209
##  6 7137926313            180
##  7 7137945446            174
##  8 7189204321            162
##  9 7207771234            159
## 10 2164458124            148
##
## # A tibble: 10 x 2
##    Hospital.affiliation.CCN.1 num_provider
##    <fct>                             <int>
##  1 <NA>                              38797
##  2 230046                             1154
##  3 360180                              804
##  4 450076                              543
##  5 230038                              423
##  6 080001                              397
##  7 330214                              383
##  8 140010                              364
##  9 050262                              357
## 10 390164                              356
##
## # A tibble: 10 x 2
##    Hospital.affiliation.LBN.1                               num_provider
##    <fct>                                                           <int>
##  1 <NA>                                                            38970
##  2 UNIVERSITY OF MICHIGAN HEALTH SYSTEM                             1154
##  3 CLEVELAND CLINIC                                                  804
##  4 UNIVERSITY OF TEXAS M D ANDERSON CANCER CENTER,THE                543
##  5 SPECTRUM HEALTH - BUTTERWORTH CAMPUS                              423
##  6 CHRISTIANA CARE HEALTH SERVICES, INC.                             397
##  7 NYU HOSPITALS CENTER                                              383
##  8 EVANSTON HOSPITAL                                                 364
##  9 RONALD REAGAN U C L A MEDICAL CENTER                              357
## 10 UPMC PRESBYTERIAN SHADYSIDE                                       356
##
## # A tibble: 10 x 2
##    Hospital.affiliation.CCN.2 num_provider
##    <fct>                             <int>
##  1 <NA>                              84816
##  2 330106                              185
##  3 050112                              153
##  4 390111                              152
##  5 390263                              149
##  6 360180                              136
##  7 390326                              131
##  8 360230                              120
##  9 450184                              119
## 10 490007                              118
##
## # A tibble: 10 x 2
##    Hospital.affiliation.LBN.2                               num_provider
##    <fct>                                                           <int>
##  1 <NA>                                                            85085
##  2 NORTH SHORE UNIVERSITY HOSPITAL                                   185
##  3 ST JOSEPH MEDICAL CENTER                                          180
##  4 ST FRANCIS HOSPITAL                                               156
##  5 SANTA MONICA - UCLA MED CTR & ORTHOPAEDIC HOSPITAL                153
##  6 HOSPITAL OF UNIV OF PENNSYLVANIA                                  152
##  7 LEHIGH VALLEY HOSPITAL - MUHLENBERG                               149
##  8 CLEVELAND CLINIC                                                  136
##  9 FAIRVIEW HOSPITAL                                                 131
## 10 ST LUKE'S HOSPITAL - ANDERSON CAMPUS                              131
##
## # A tibble: 10 x 2
```

```
## # A tibble: 10 x 2
##    Hospital.affiliation.CCN.3 num_provider
##    <fct>                             <int>
##  1 <NA>                             107369
##  2 110008                               82
##  3 360364                               79
##  4 520189                               71
##  5 240001                               68
##  6 490007                               62
##  7 360077                               60
##  8 490046                               60
##  9 520139                               55
## 10 150024                               52
##
## # A tibble: 10 x 2
##    Hospital.affiliation.LBN.3       num_provider
##    <fct>                                   <int>
##  1 <NA>                                   107595
##  2 NORTHSIDE HOSPITAL CHEROKEE                82
##  3 AURORA MEDICAL CTR KENOSHA                 71
##  4 AURORA MEDICAL CENTER                      68
##  5 NORTH MEMORIAL MEDICAL CENTER              68
##  6 FAIRVIEW HOSPITAL                          63
##  7 SENTARA NORFOLK GENERAL HOSPITAL           62
##  8 GOOD SAMARITAN HOSPITAL                    61
##  9 SENTARA LEIGH HOSPITAL                     60
## 10 ST JOSEPH'S HOSPITAL                       59
##
## # A tibble: 10 x 2
##    Hospital.affiliation.CCN.4 num_provider
##    <fct>                             <int>
##  1 <NA>                             118454
##  2 050537                               57
##  3 360230                               53
##  4 520206                               48
##  5 490119                               45
##  6 260062                               41
##  7 450068                               38
##  8 360077                               37
##  9 490046                               36
## 10 150157                               34
##
## # A tibble: 10 x 2
##    Hospital.affiliation.LBN.4            num_provider
##    <fct>                                        <int>
##  1 <NA>                                        118583
##  2 AURORA MEDICAL CENTER                           77
##  3 SUTTER DAVIS HOSPITAL                           57
##  4 HILLCREST HOSPITAL                              53
##  5 SENTARA PRINCESS ANNE HOSPITAL                  45
##  6 SAINT LUKES NORTHLAND HOSPITAL                  41
##  7 ST ANTHONY HOSPITAL                             41
##  8 GOOD SAMARITAN HOSPITAL                         39
##  9 MEMORIAL HERMANN TEXAS MEDICAL CENTER           38
## 10 FAIRVIEW HOSPITAL                               37
##
## # A tibble: 10 x 2
##    Hospital.affiliation.CCN.5 num_provider
##    <fct>                             <int>
##  1 <NA>                             123981
##  2 520207                               52
##  3 520139                               49
##  4 520206                               43
##  5 340060                               38
##  6 100314                               29
##  7 500077                               27
##  8 360230                               25
##  9 390183                               25
## 10 360143                               24
##
## # A tibble: 10 x 2
##    Hospital.affiliation.LBN.5         num_provider
##    <fct>                                     <int>
```

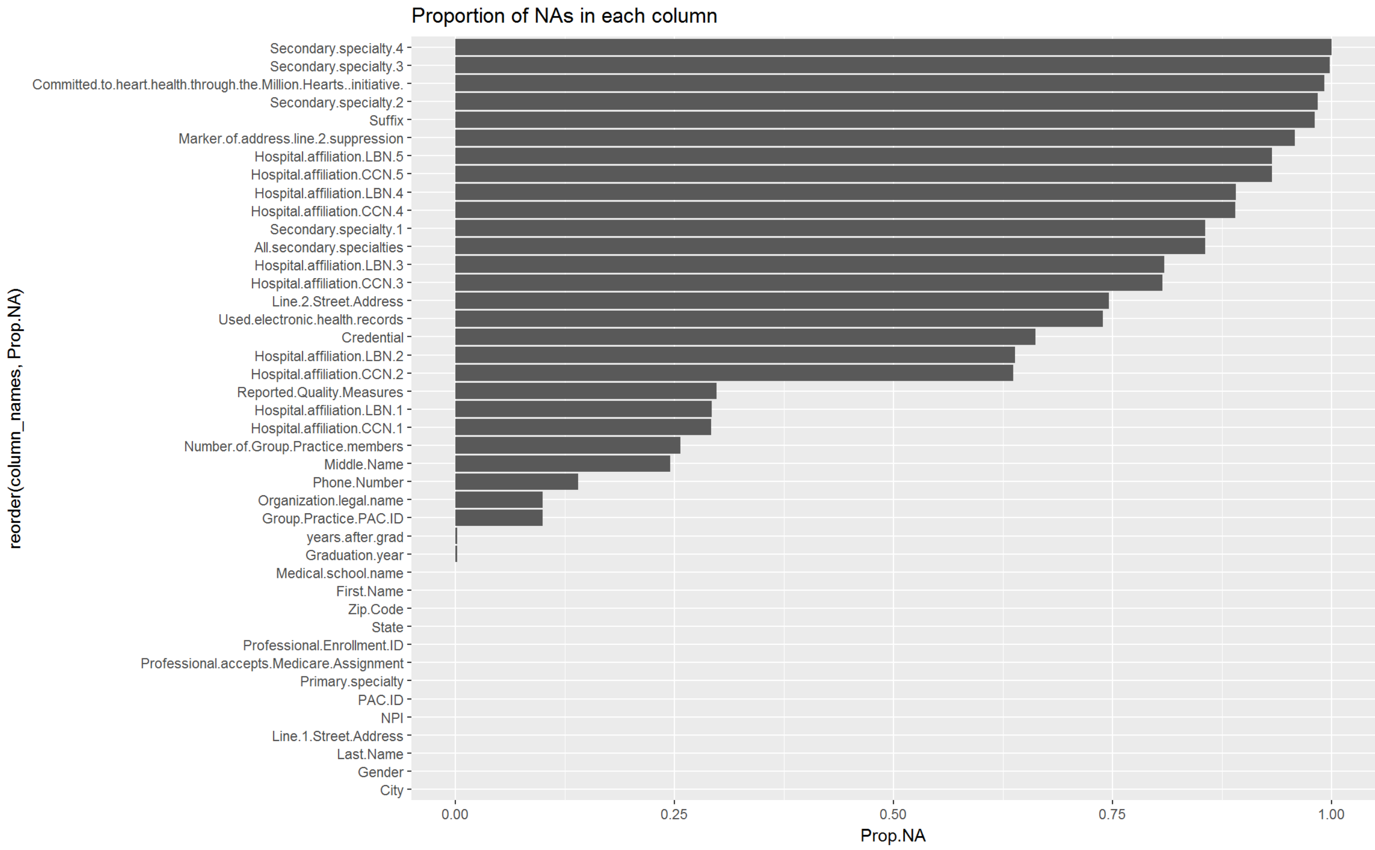
```
##   1 <NA>                                     124060
##   2 AURORA MEDICAL CENTER                        95
##   3 AURORA WEST ALLIS MEDICAL CENTER             49
##   4 MOREHEAD MEMORIAL HOSPITAL                   38
##   5 DOCTORS HOSPITAL                             31
##   6 WEST KENDALL BAPTIST HOSPITAL                29
##   7 PROVIDENCE HOLY FAMILY HOSPITAL              27
##   8 HILLCREST HOSPITAL                           25
##   9 MEMORIAL HOSPITAL                            25
##  10 ST LUKE'S MINERS MEMORIAL HOSPITAL           25
## 
## # A tibble: 2 x 2
##   Professional.accepts.Medicare.Assignment num_provider
##   <fct>                                           <int>
## 1 Y                                              128582
## 2 M                                                4479
## 
## # A tibble: 2 x 2
##   Reported.Quality.Measures num_provider
##   <fct>                            <int>
## 1 Y                                93356
## 2 <NA>                             39705
## 
## # A tibble: 2 x 2
##   Used.electronic.health.records num_provider
##   <fct>                                 <int>
## 1 <NA>                                  98384
## 2 Y                                     34677
## 
## # A tibble: 2 x 2
##   Committed.to.heart.health.through.the.Million.Hearts..init~ num_provider
##   <fct>                                                              <int>
## 1 <NA>                                                              131953
## 2 Y                                                                   1108
## 
## [1] "years.after.grad"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##    0.00   10.00   19.00   20.24   29.00   70.00     330
```

There are many NA values. How many NA are in each columns? How many providers have a lot of NA values?

```
col.NA <-
  provider %>%
  summarise_all(funs(
    signif(
      sum(is.na(.)) / n(),
      digits = 3)))

col.NA <-
  as_tibble(cbind(column_names = names(col.NA), t(col.NA)))

names(col.NA)[2] <- c('Prop.NA')
col.NA$Prop.NA <- as.numeric(col.NA$Prop.NA)

col.NA <-
  col.NA %>%
  arrange(desc(Prop.NA)) %>%
  print(n = Inf)
```

```
## # A tibble: 42 x 2
##    column_names                                                  Prop.NA
##    <chr>                                                           <dbl>
##  1 Secondary.specialty.4                                         1.00e+0
##  2 Secondary.specialty.3                                         9.98e-1
##  3 Committed.to.heart.health.through.the.Million.Hearts..initia~ 9.92e-1
##  4 Secondary.specialty.2                                         9.84e-1
##  5 Suffix                                                        9.81e-1
##  6 Marker.of.address.line.2.suppression                          9.58e-1
##  7 Hospital.affiliation.CCN.5                                    9.32e-1
##  8 Hospital.affiliation.LBN.5                                    9.32e-1
##  9 Hospital.affiliation.LBN.4                                    8.91e-1
## 10 Hospital.affiliation.CCN.4                                    8.90e-1
## 11 Secondary.specialty.1                                         8.56e-1
## 12 All.secondary.specialties                                     8.56e-1
## 13 Hospital.affiliation.LBN.3                                    8.09e-1
## 14 Hospital.affiliation.CCN.3                                    8.07e-1
## 15 Line.2.Street.Address                                         7.46e-1
## 16 Used.electronic.health.records                                7.39e-1
## 17 Credential                                                    6.62e-1
## 18 Hospital.affiliation.LBN.2                                    6.39e-1
## 19 Hospital.affiliation.CCN.2                                    6.37e-1
## 20 Reported.Quality.Measures                                     2.98e-1
## 21 Hospital.affiliation.LBN.1                                    2.93e-1
## 22 Hospital.affiliation.CCN.1                                    2.92e-1
## 23 Number.of.Group.Practice.members                              2.57e-1
## 24 Middle.Name                                                   2.45e-1
## 25 Phone.Number                                                  1.40e-1
## 26 Organization.legal.name                                       1.00e-1
## 27 Group.Practice.PAC.ID                                         1.00e-1
## 28 Graduation.year                                               2.48e-3
## 29 years.after.grad                                              2.48e-3
## 30 First.Name                                                    7.52e-6
## 31 Medical.school.name                                           7.52e-6
## 32 NPI                                                           0.
## 33 PAC.ID                                                        0.
## 34 Professional.Enrollment.ID                                    0.
## 35 Last.Name                                                     0.
## 36 Gender                                                        0.
## 37 Primary.specialty                                             0.
## 38 Line.1.Street.Address                                         0.
## 39 City                                                          0.
## 40 State                                                         0.
## 41 Zip.Code                                                      0.
## 42 Professional.accepts.Medicare.Assignment                      0.
```

```
ggplot(col.NA, aes(x = reorder(column_names,Prop.NA), y = Prop.NA)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Proportion of NAs in each column")
```



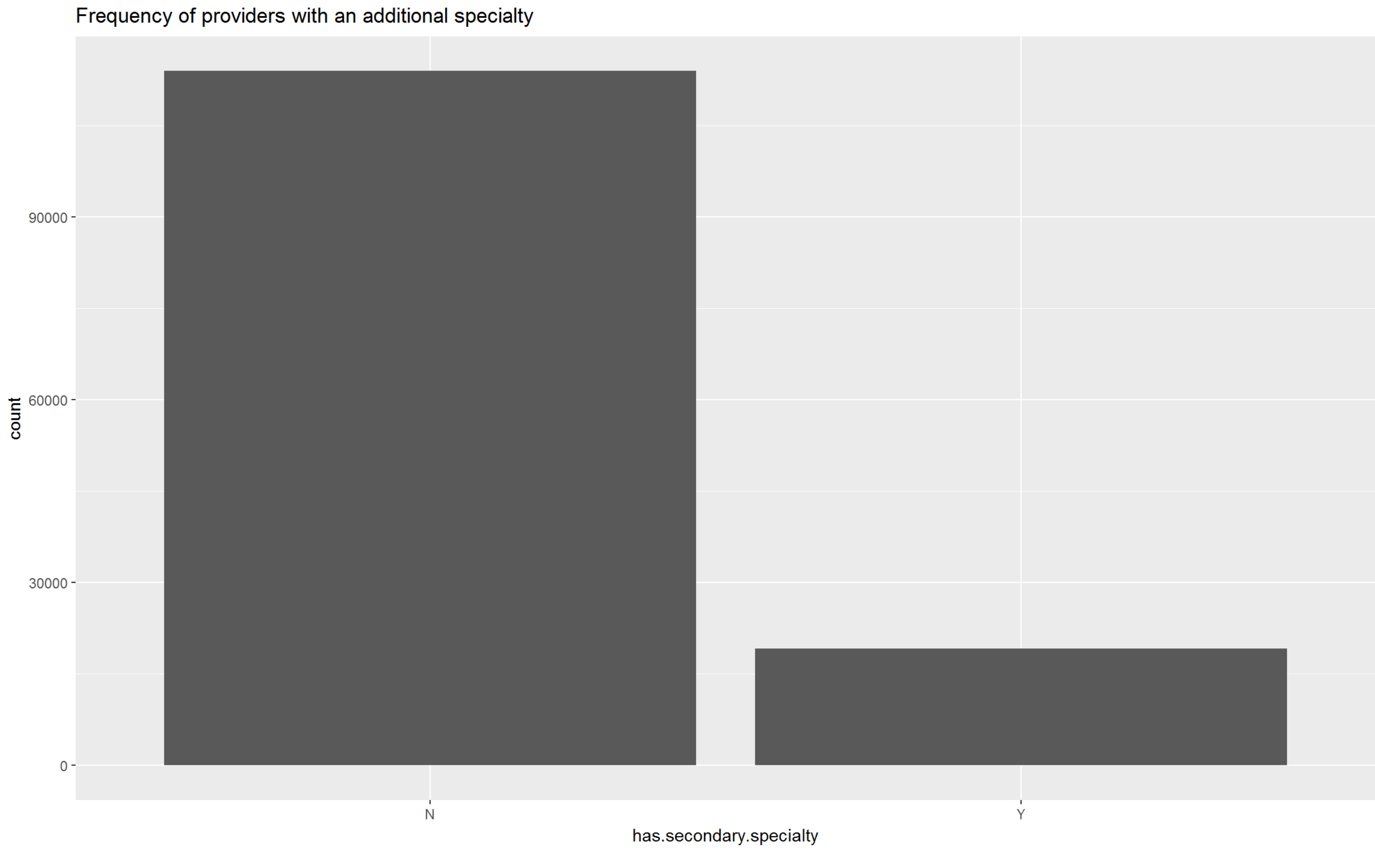
There are a bunch of columns that have NA, but not all of them are important. For example, `Suffix` and `Secondary.specialty.4` are not important to have complete information. Doing some feature engineering to consolidate information, such as specialties and hospital affiliations.

Creating the variable `has.secondary.specialty` to consolidate the secondary specialty columns.

```
provider <-
  provider %>%
  mutate(has.secondary.specialty =
           factor(ifelse(is.na(All.secondary.specialties), 'N', 'Y')))

provider %>%
  group_by(has.secondary.specialty) %>%
  dplyr::summarise(n() / nrow(provider))
```

```
## # A tibble: 2 x 2
##   has.secondary.specialty `n()/nrow(provider)`
##   <fct>                                  <dbl>
## 1 N                                      0.856
## 2 Y                                      0.144
```

```
ggplot(provider,aes(x = has.secondary.specialty)) +
  geom_bar() +
  labs(title = "Frequency of providers with an additional specialty")
```



85.6% of the providers in this table do not have a secondary specialty and 14.4% do. Can now remove those columns about Secondary Specialty. This was somewhat surprising to see.

```
col.remove = c("Secondary.specialty.1", "Secondary.specialty.2",
               "Secondary.specialty.3", "Secondary.specialty.4",
               "All.secondary.specialties")

# Add columns that we engineered above

col.remove <- list.append(col.remove, "Graduation.year")

cat("These columns are still being considered for analysis: \n",
    paste(setdiff(names(provider),col.remove), collapse = ' \n '))
```

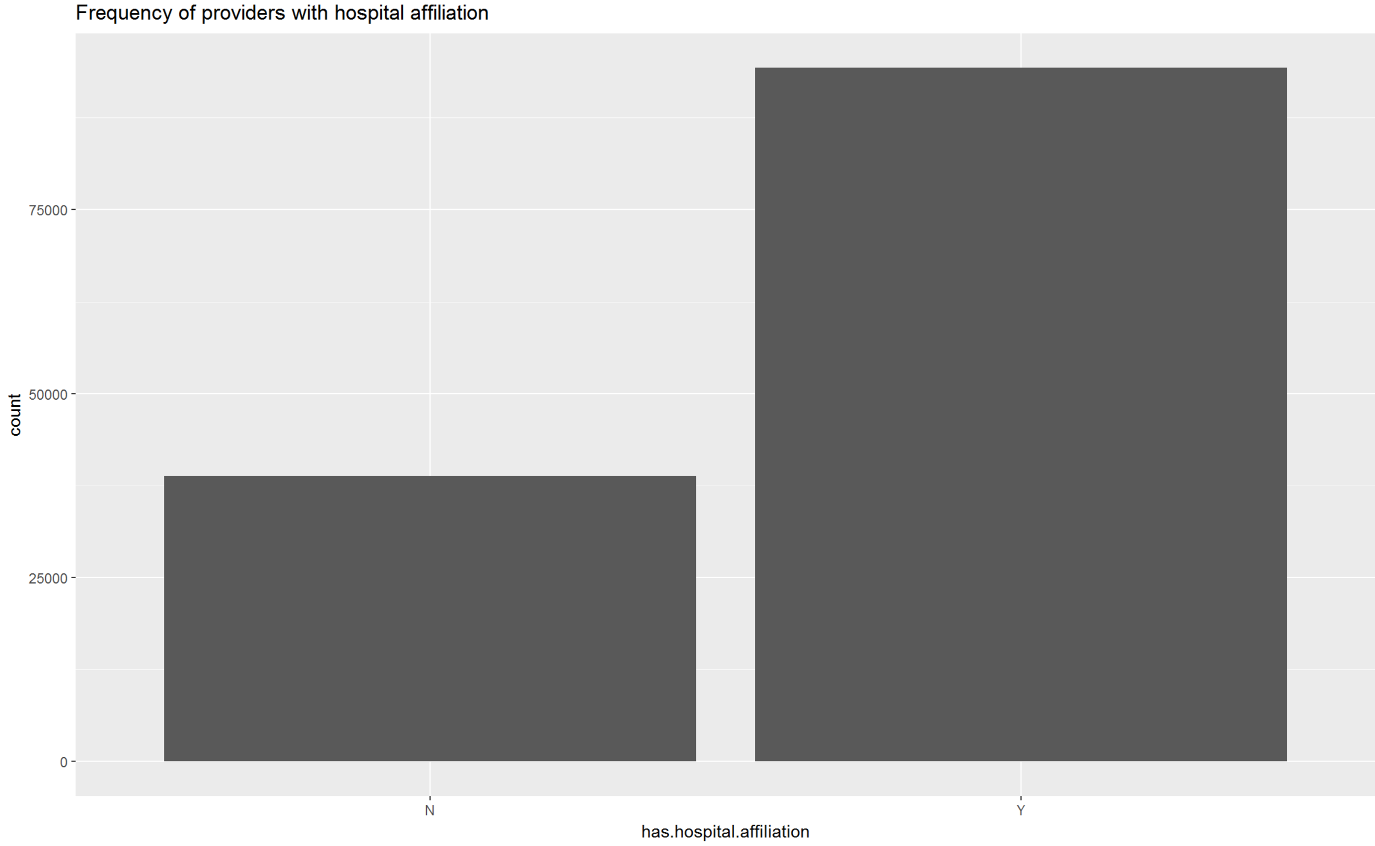
```
## These columns are still being considered for analysis:
##   NPI
##   PAC.ID
##   Professional.Enrollment.ID
##   Last.Name
##   First.Name
##   Middle.Name
##   Suffix
##   Gender
##   Credential
##   Medical.school.name
##   Primary.specialty
##   Organization.legal.name
##   Group.Practice.PAC.ID
##   Number.of.Group.Practice.members
##   Line.1.Street.Address
##   Line.2.Street.Address
##   Marker.of.address.line.2.suppression
##   City
##   State
##   Zip.Code
##   Phone.Number
##   Hospital.affiliation.CCN.1
##   Hospital.affiliation.LBN.1
##   Hospital.affiliation.CCN.2
##   Hospital.affiliation.LBN.2
##   Hospital.affiliation.CCN.3
##   Hospital.affiliation.LBN.3
##   Hospital.affiliation.CCN.4
##   Hospital.affiliation.LBN.4
##   Hospital.affiliation.CCN.5
##   Hospital.affiliation.LBN.5
##   Professional.accepts.Medicare.Assignment
##   Reported.Quality.Measures
##   Used.electronic.health.records
##   Committed.to.heart.health.through.the.Million.Hearts..initiative.
##   years.after.grad
##   has.secondary.specialty
```

Consolidate hospital affiliation information into `has.hospital.affiliation`.

```
provider <-
  provider %>%
  mutate(has.hospital.affiliation =
           factor(ifelse((is.na(Hospital.affiliation.CCN.1) &
                            is.na(Hospital.affiliation.CCN.2) &
                            is.na(Hospital.affiliation.CCN.3) &
                            is.na(Hospital.affiliation.CCN.4) &
                            is.na(Hospital.affiliation.CCN.5)), 'N','Y')))

provider %>%
  group_by(has.hospital.affiliation) %>%
  dplyr::summarise(n() / nrow(provider))
```

```
## # A tibble: 2 x 2
##   has.hospital.affiliation `n()/nrow(provider)`
##   <fct>                                    <dbl>
## 1 N                                        0.292
## 2 Y                                        0.708
```

```
ggplot(provider,aes(x = has.hospital.affiliation)) +
  geom_bar() +
  labs(title = "Frequency of providers with hospital affiliation")
```



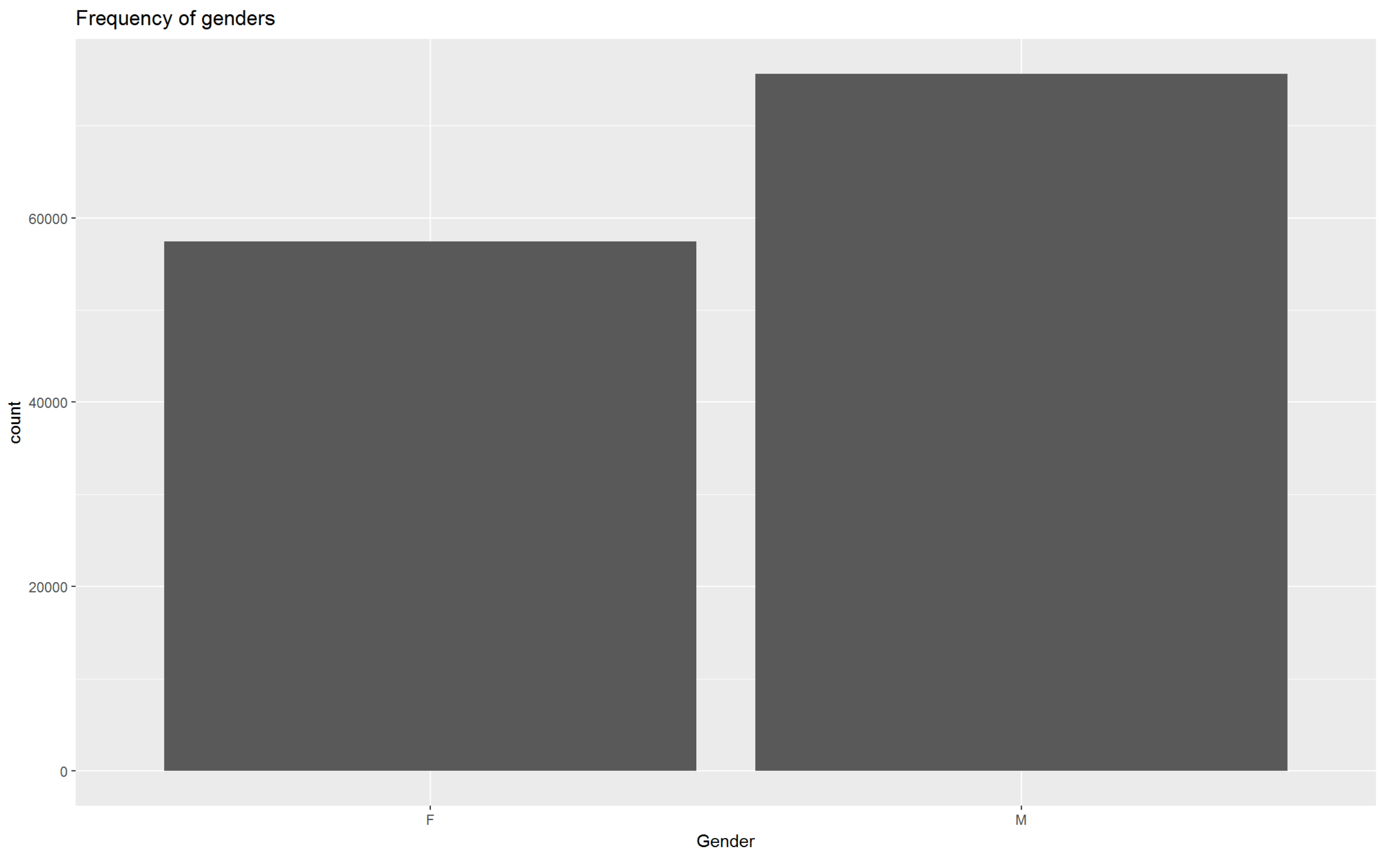
This was also surprising to see because I expected more providers to operate within their own primary practices.

```
col.remove <- list.append(col.remove, c("Hospital.affiliation.CCN.1",
                                        "Hospital.affiliation.CCN.2",
                                        "Hospital.affiliation.CCN.3",
                                        "Hospital.affiliation.CCN.4",
                                        "Hospital.affiliation.CCN.5",
                                        "Hospital.affiliation.LBN.1",
                                        "Hospital.affiliation.LBN.2",
                                        "Hospital.affiliation.LBN.3",
                                        "Hospital.affiliation.LBN.4",
                                        "Hospital.affiliation.LBN.5"))

cat("These columns are still being considered for analysis: \n",
    paste(setdiff(names(provider),col.remove), collapse = ' \n '))
```

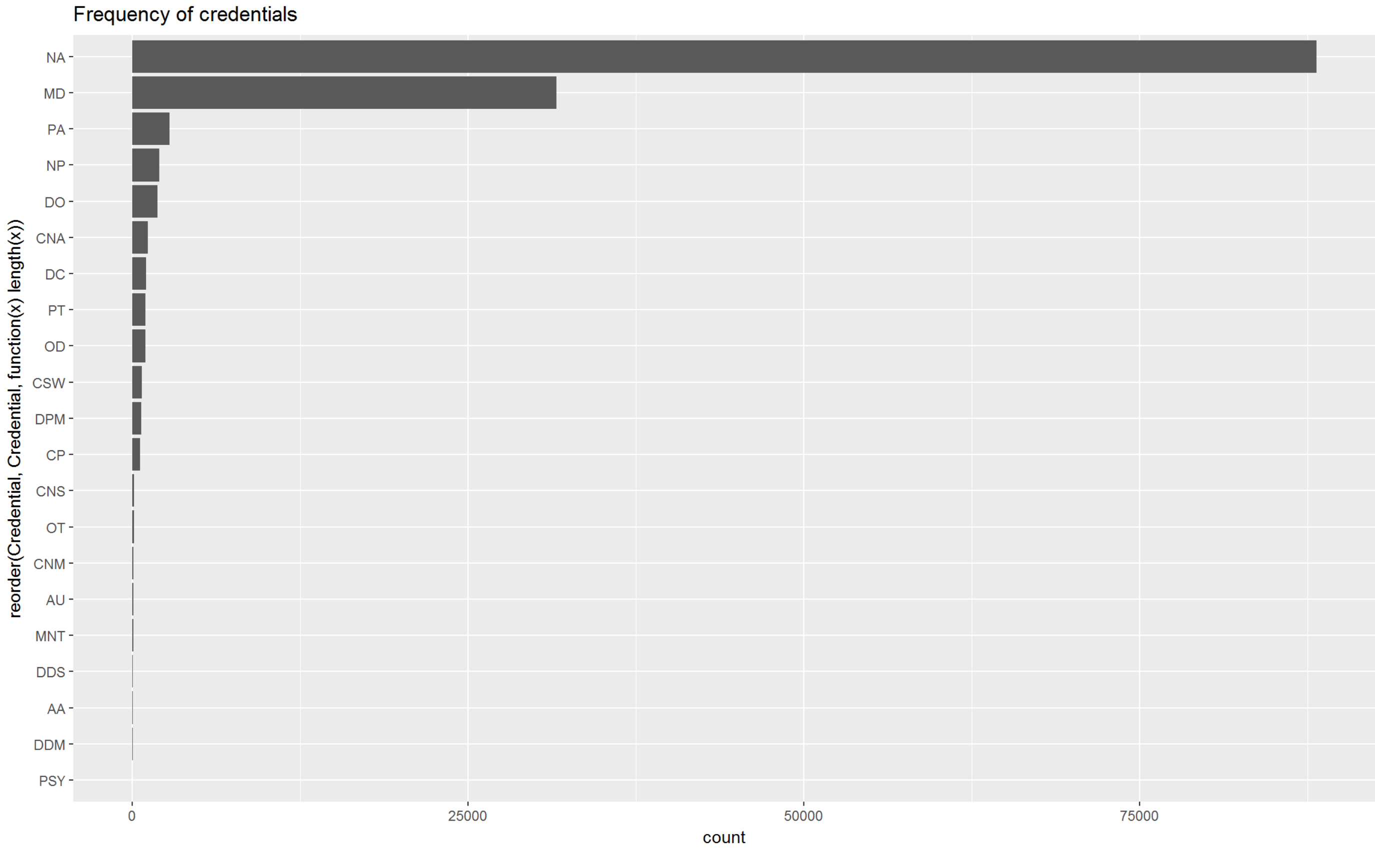
```
## These columns are still being considered for analysis:
##  NPI
##  PAC.ID
##  Professional.Enrollment.ID
##  Last.Name
##  First.Name
##  Middle.Name
##  Suffix
##  Gender
##  Credential
##  Medical.school.name
##  Primary.specialty
##  Organization.legal.name
##  Group.Practice.PAC.ID
##  Number.of.Group.Practice.members
##  Line.1.Street.Address
##  Line.2.Street.Address
##  Marker.of.address.line.2.suppression
##  City
##  State
##  Zip.Code
##  Phone.Number
##  Professional.accepts.Medicare.Assignment
##  Reported.Quality.Measures
##  Used.electronic.health.records
##  Committed.to.heart.health.through.the.Million.Hearts..initiative.
##  years.after.grad
##  has.secondary.specialty
##  has.hospital.affiliation
```

Create frequency plots for the other variables if low number of unique values

```r
# Gender
ggplot(provider, aes(x = Gender)) +
  geom_bar()  +
  labs(title = "Frequency of genders")
```

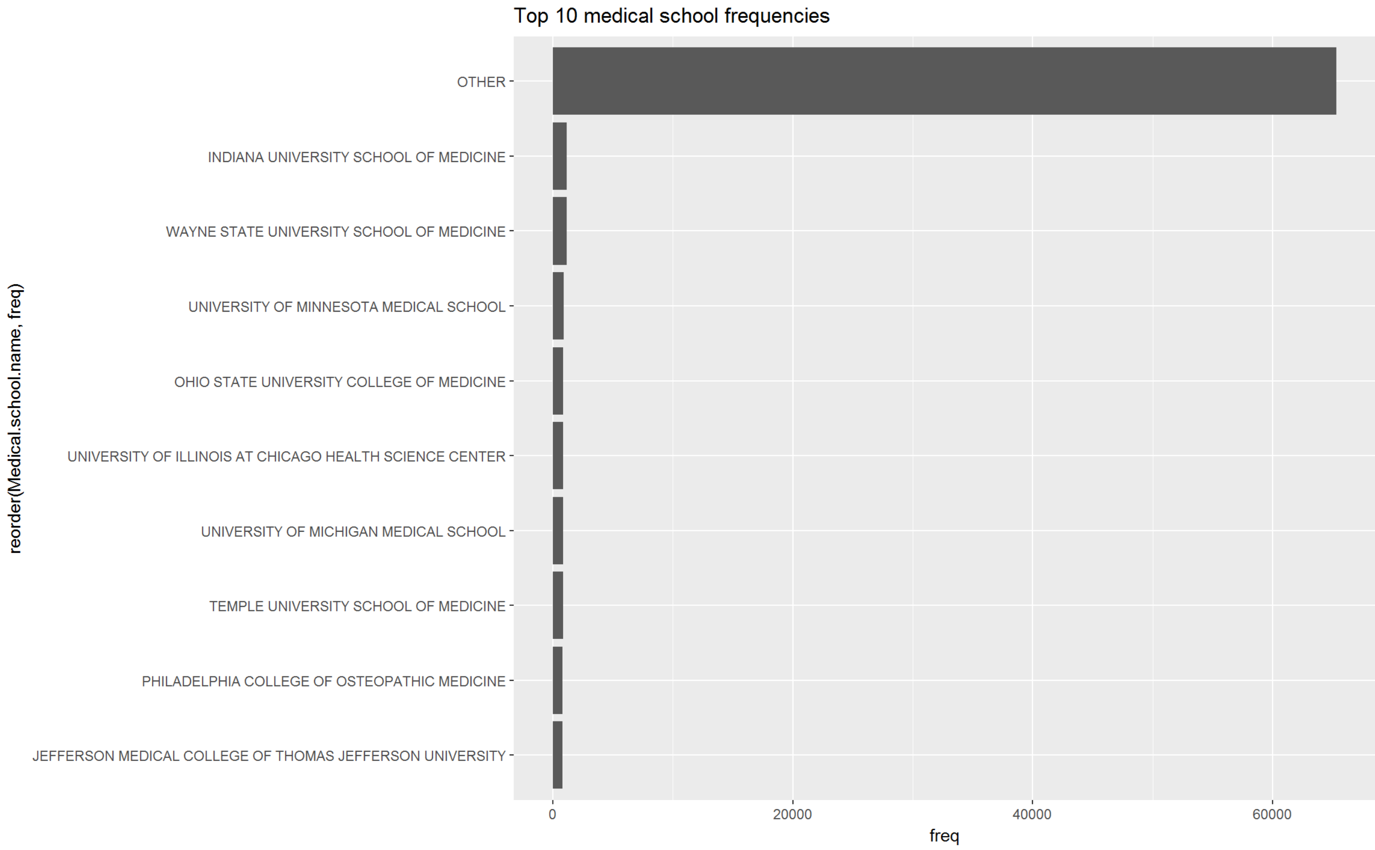


```r
# Credential
ggplot(provider, aes(x = reorder(Credential,Credential,
                                 function(x)length(x)))) +
  geom_bar()  +
  labs(title = "Frequency of credentials") +
  coord_flip()
```



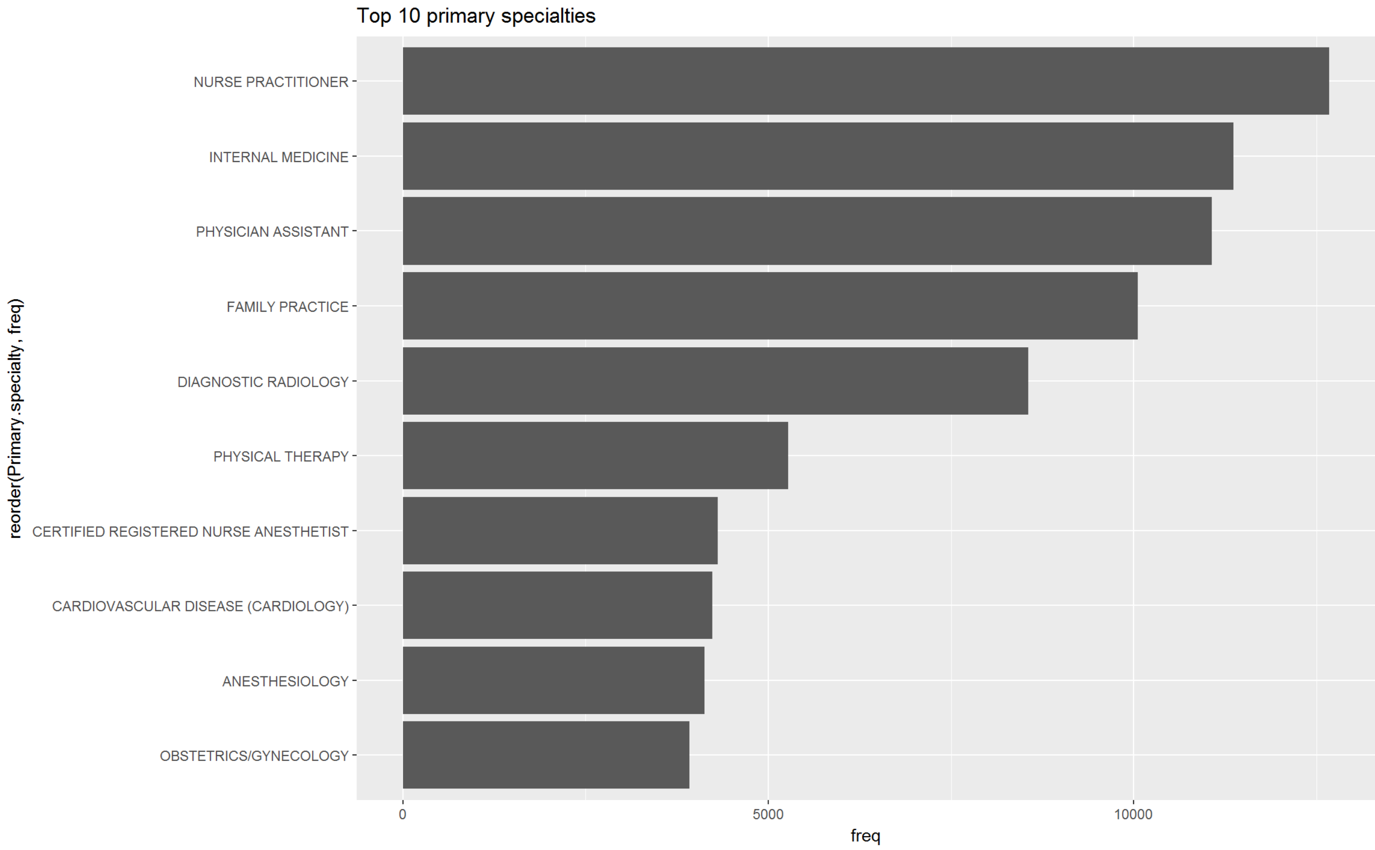
Many providers have `NA` for the credentials, which is very strange. I would like to compare providers with no credential (population 1) with providers with credentials (population 2) to see if these populations differ at all.

```r
# [TODO] Complete hypothesis testing

# Medical School Name
provider %>%
  group_by(Medical.school.name) %>%
  dplyr::summarise(freq = n()) %>%
  dplyr::arrange(desc(freq)) %>%
  top_n(n = 10,freq) %>%
  ggplot(aes(x = reorder(Medical.school.name,freq), y = freq)) +
  geom_bar(stat = 'identity') +
  coord_flip() +
  labs(title = "Top 10 medical school frequencies")
```



Due to the high number of unique values in `Medical.school.name`, this variable may need to be binned into a lower dimensional value set.

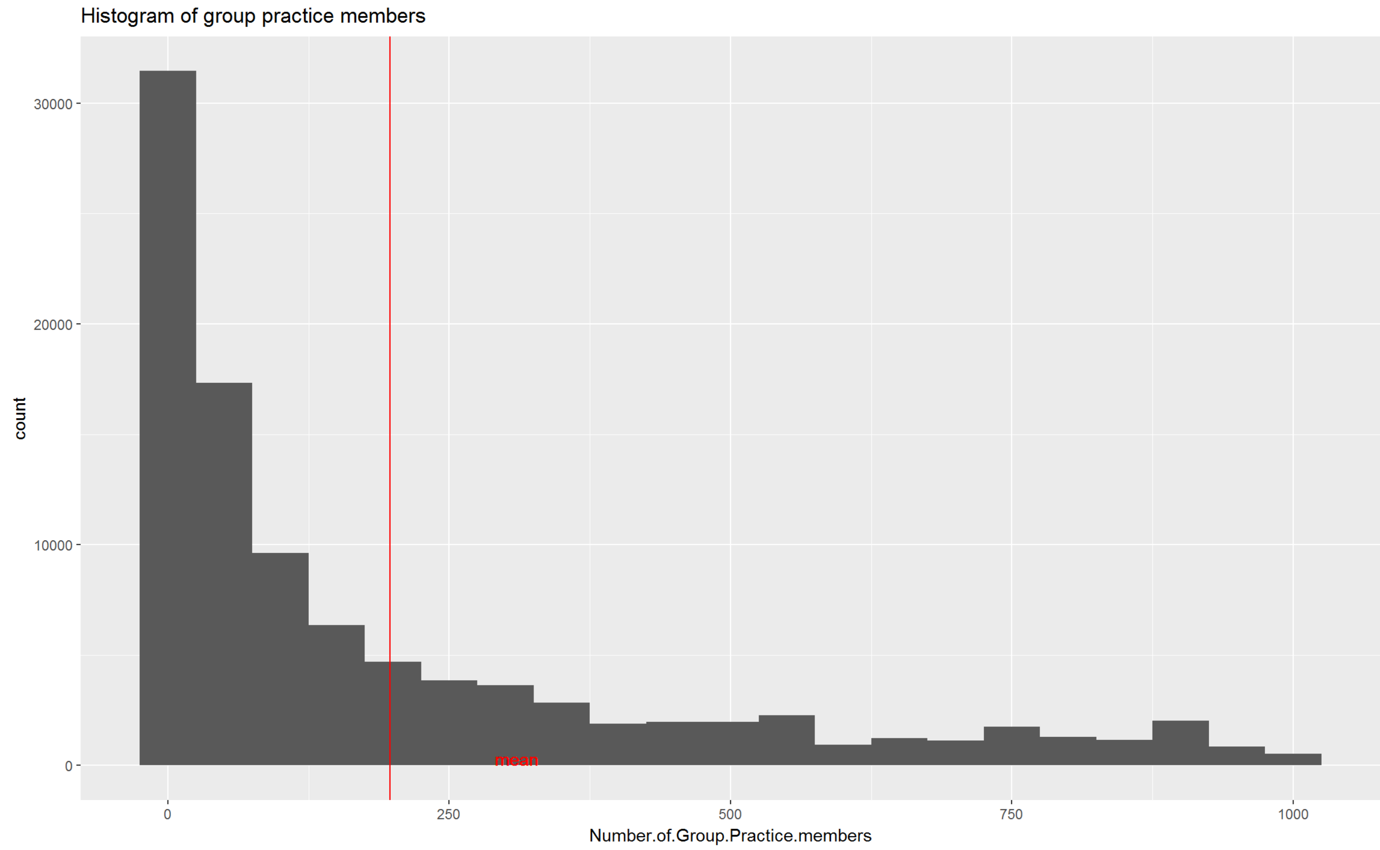
```
topthird.medical.school <-
  provider %>%
  group_by(Medical.school.name) %>%
  dplyr::summarise(freq = n()) %>%
  dplyr::arrange(desc(freq)) %>%
  top_n(n = round(.34 * length(unique(provider$Medical.school.name))),freq) %>%
  dplyr::select(Medical.school.name)

tailthird.medical.school <-
  provider %>%
  group_by(Medical.school.name) %>%
  dplyr::summarise(freq = n()) %>%
  dplyr::arrange(desc(freq)) %>%
  top_n(n = -round(.33 * length(unique(provider$Medical.school.name))),freq) %>%
  dplyr::select(Medical.school.name)


# [TODO] Repeat for the other variables with multiple factors

# Primary Specialty
provider %>%
  group_by(Primary.specialty) %>%
  dplyr::summarise(freq = n()) %>%
  dplyr::arrange(desc(freq)) %>%
  top_n(n = 10,freq) %>%
  ggplot(aes(x = reorder(Primary.specialty,freq),y = freq)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Top 10 primary specialties")
```



Binned the lowest third of `Primary.specialty` into a “Rare” bucket

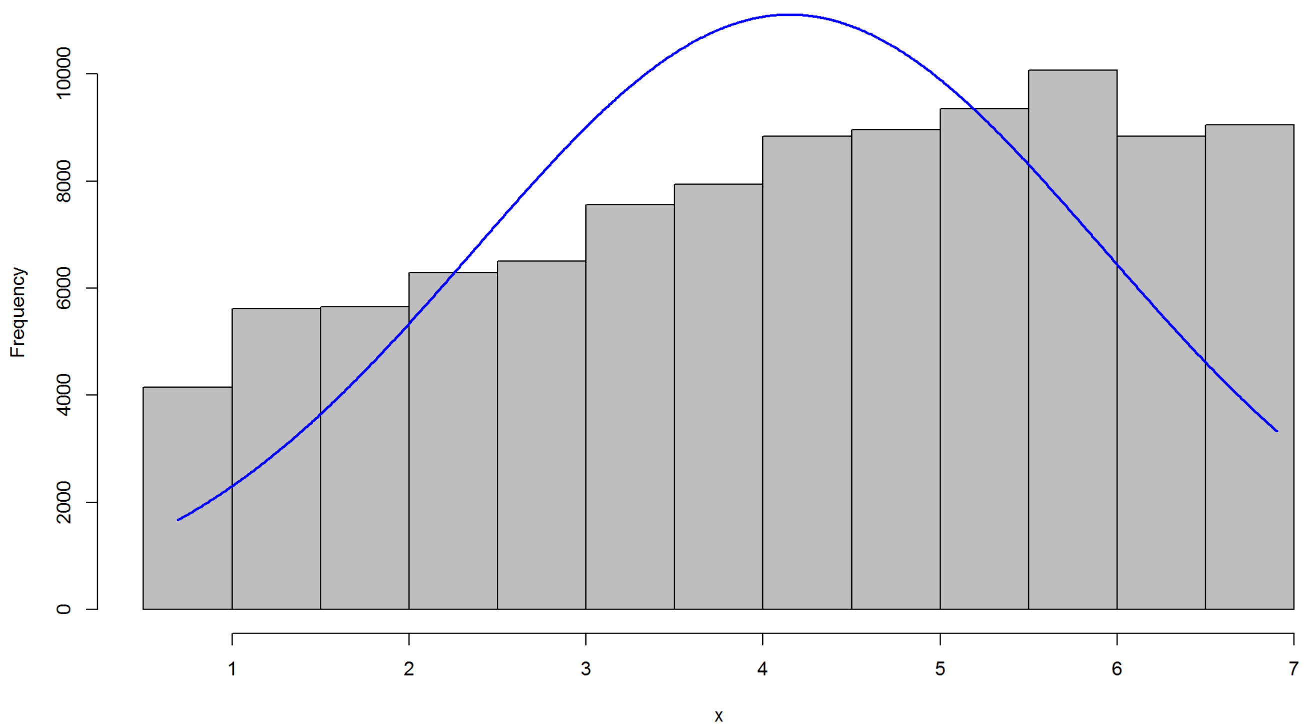
```
tailthird.specialty <-
  provider %>%
  group_by(Primary.specialty) %>%
  dplyr::summarise(freq = n()) %>%
  dplyr::arrange(desc(freq)) %>%
  top_n(n = -round(.33 * length(unique(provider$Primary.specialty))),freq) %>%
  dplyr::select(Primary.specialty)
```

```
# Group Practice Members
provider %>%
  filter(!is.na(Number.of.Group.Practice.members)) %>%
  ggplot(aes(x = Number.of.Group.Practice.members)) +
  geom_histogram(binwidth = 50) +
  labs(title = "Histogram of group practice members") +
  geom_vline(aes(xintercept = mean(Number.of.Group.Practice.members)),
             colour = "red") +
  geom_text(aes(x=310, label="mean", y=300), colour="red")
```

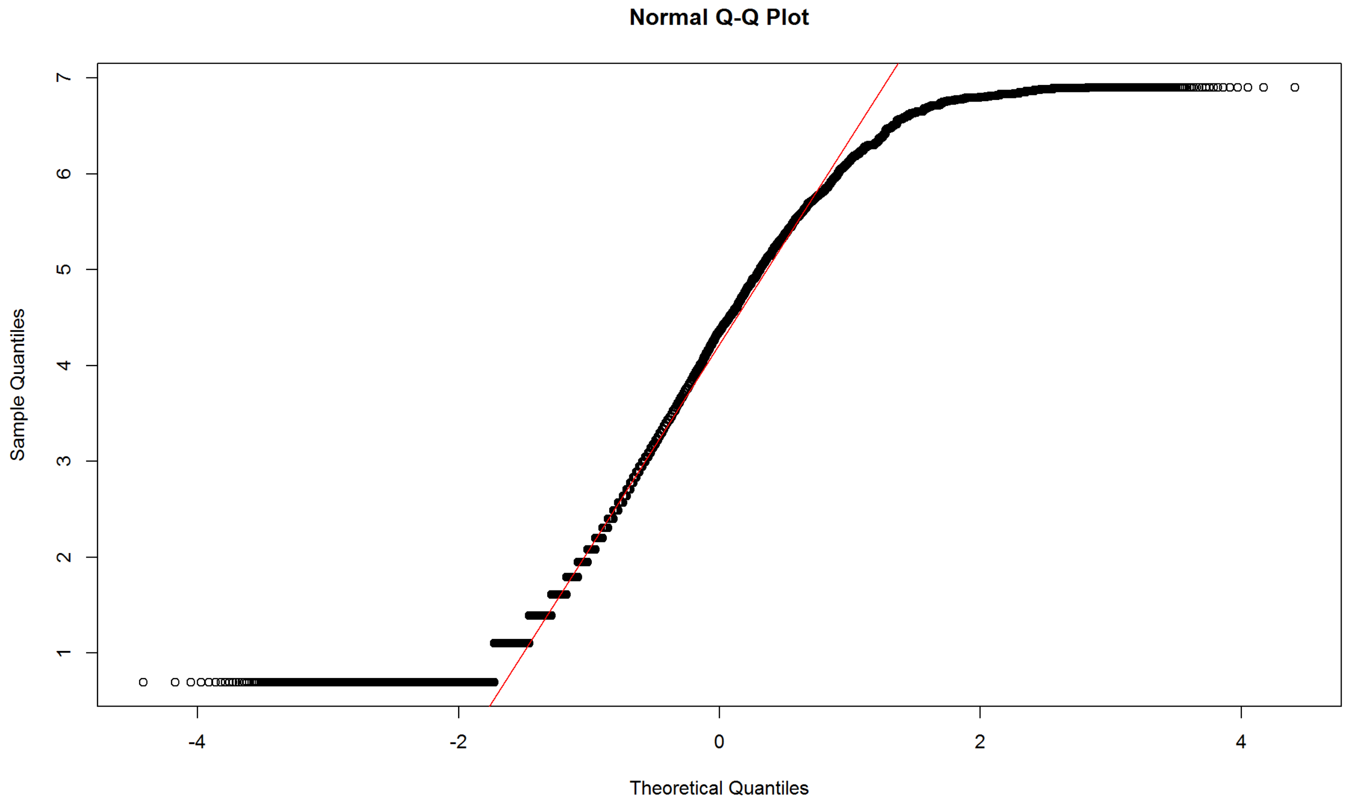


Need to normalize this data. Lognormal was chosen to normalize this data over boxcox due to the computational simplicity and satisfactory results.
Refer to this document

```
provider$Number.of.Group.Practice.members_log <-
  log(provider$Number.of.Group.Practice.members)
plotNormalHistogram(provider$Number.of.Group.Practice.members_log)
```



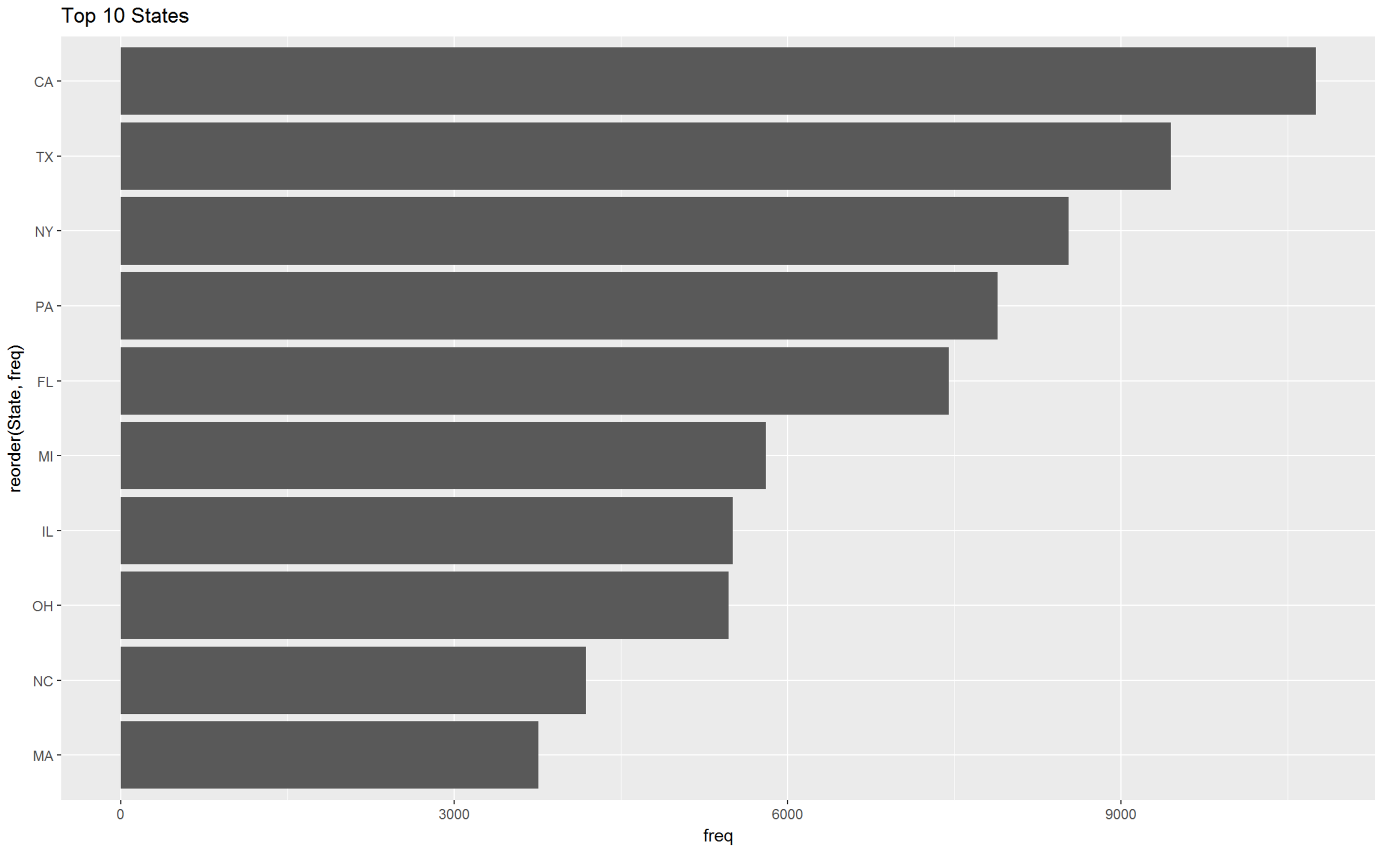

```
qqnorm(provider$Number.of.Group.Practice.members_log)
qqline(provider$Number.of.Group.Practice.members_log, col = 'red')
```




```
col.remove <- list.append(col.remove, 'Number.of.Group.Practice.members')
```

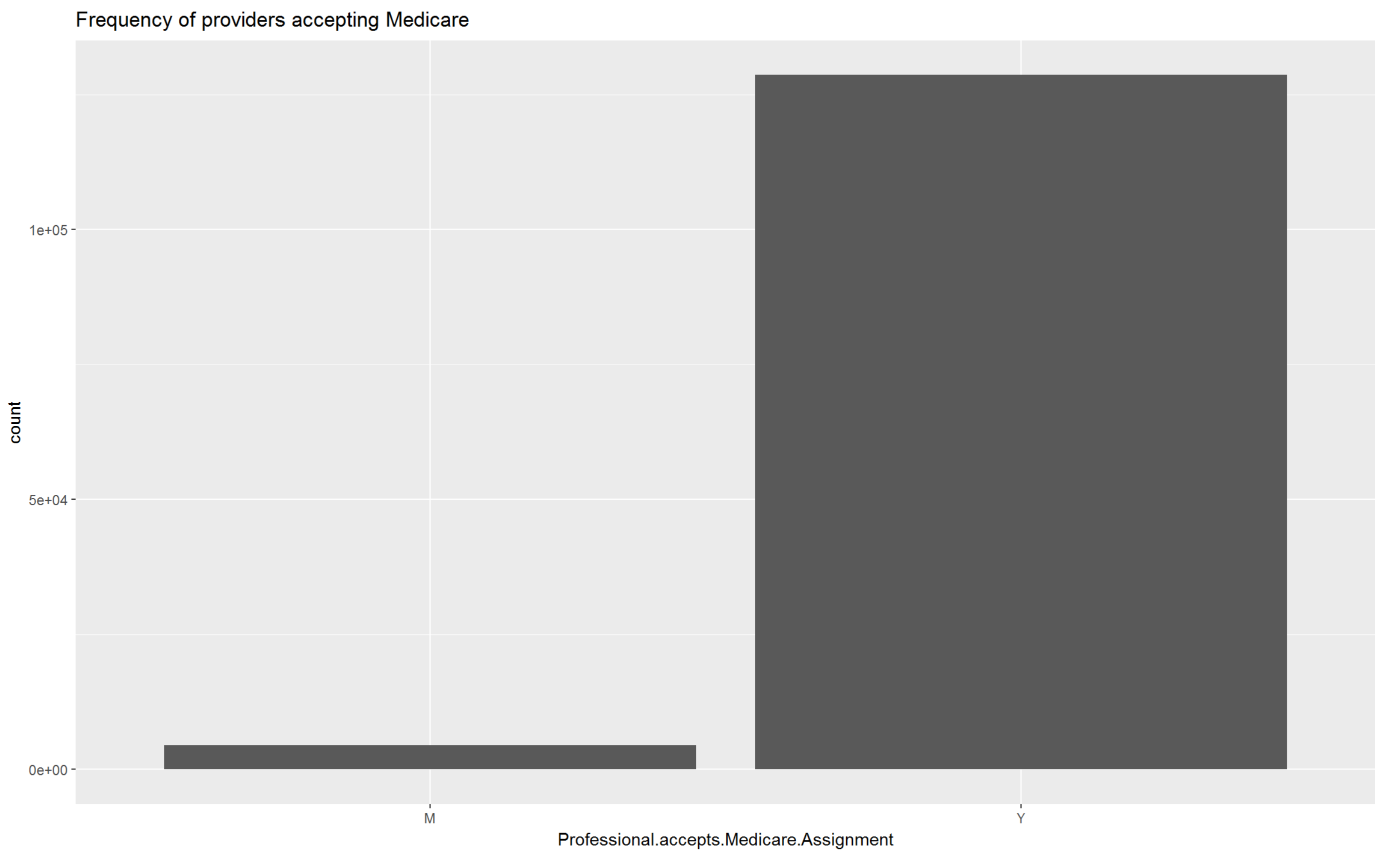
Looking at the Q-Q Plot of log normalized `Number.of.Group.Practice.members`, you can easily see that between [-1,1] theoretical quantiles, the data fits a normal distribution.

```
# State
provider %>%
  group_by(State) %>%
  dplyr::summarise(freq = n()) %>%
  dplyr::arrange(desc(freq)) %>%
  top_n(n = 10,freq) %>%
  ggplot(aes(x = reorder(State,freq),y = freq)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Top 10 States")
```



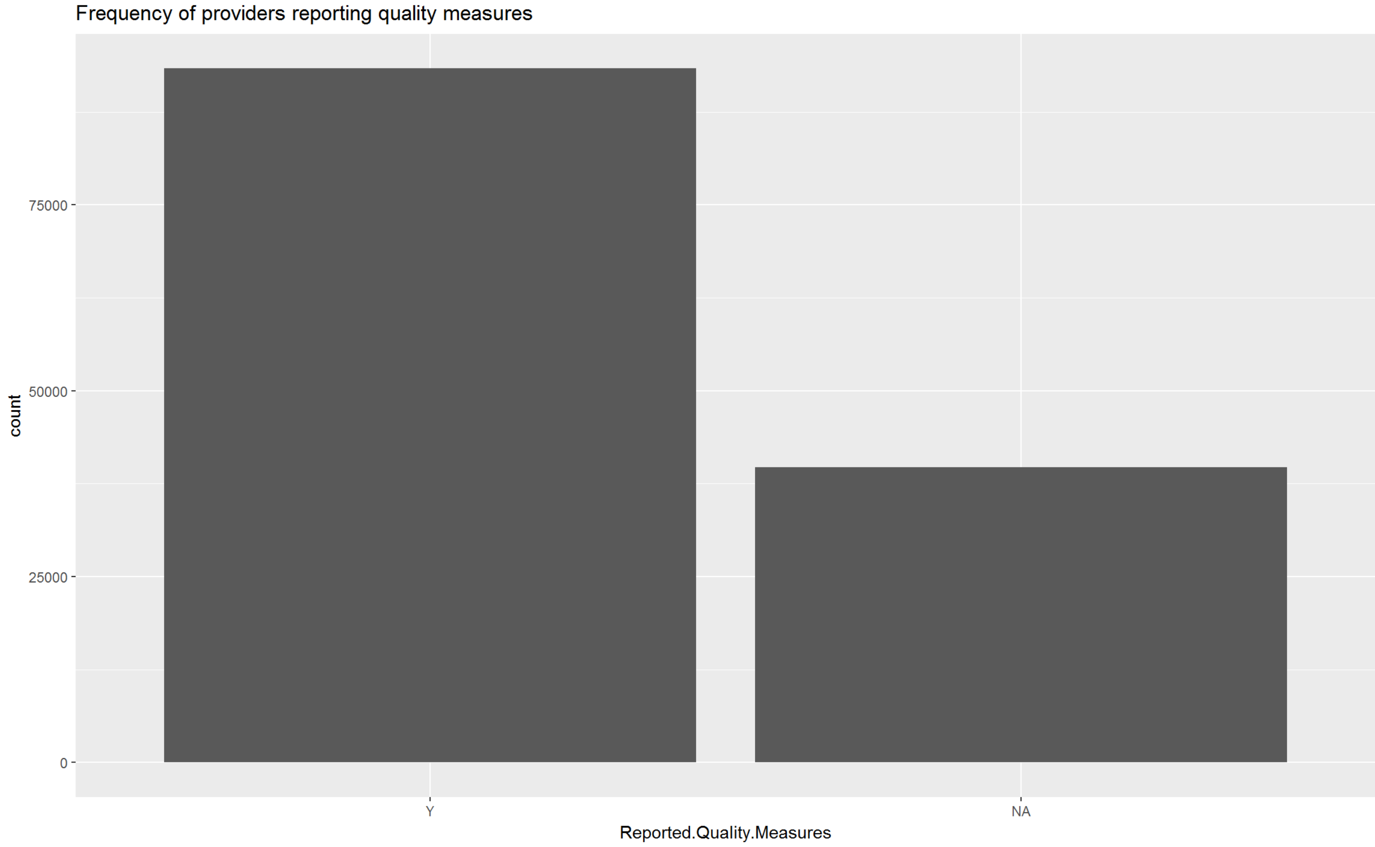
CA, TX, and NY round out the top 3 states, which makes sense because they are some of the largest states with densely populated cities.

```
# Location

# Accepts Medicare
ggplot(provider,aes(x = Professional.accepts.Medicare.Assignment)) +
  geom_bar() +
  labs(title = "Frequency of providers accepting Medicare")
```



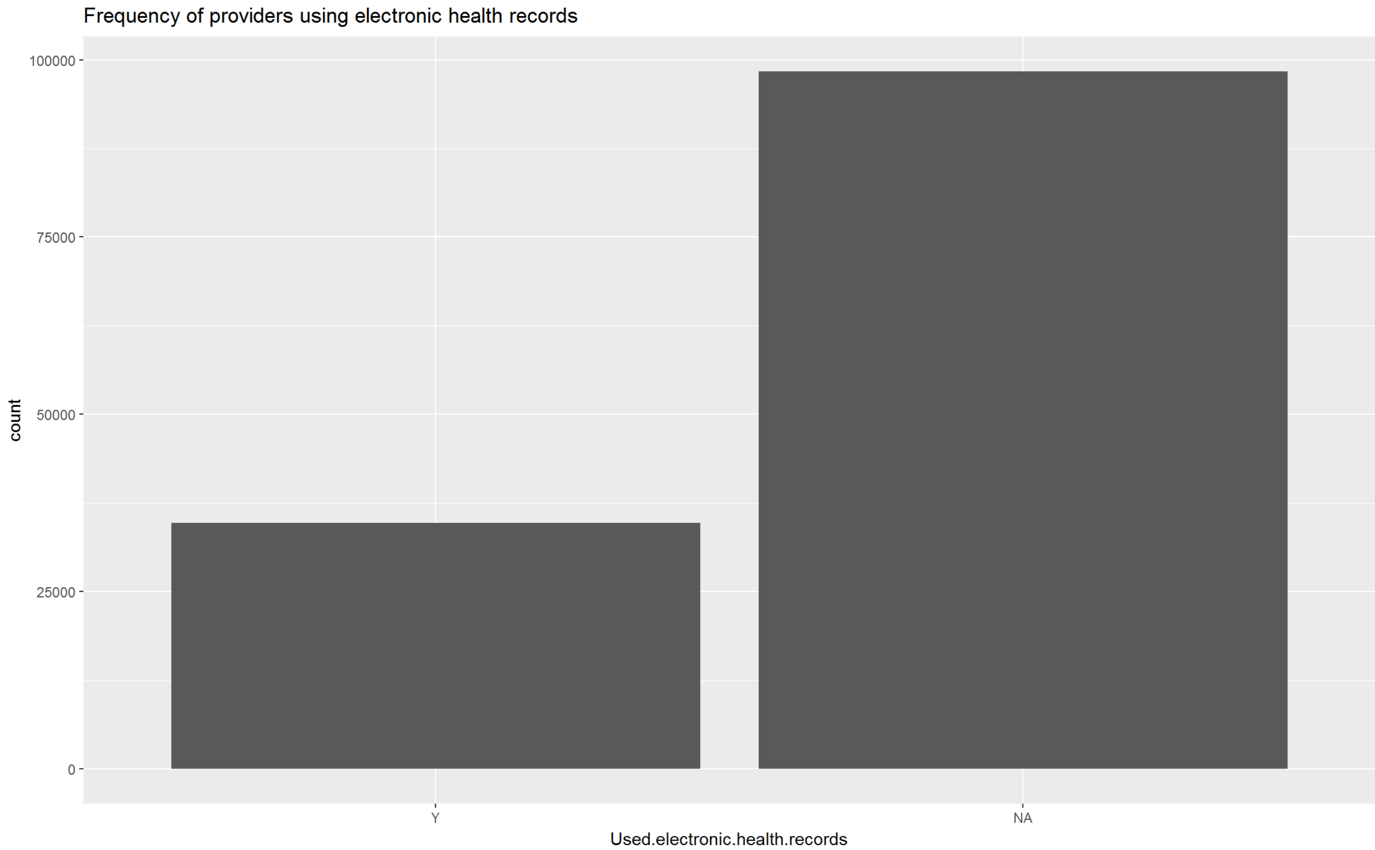
value of ‘M’ indicates a provider *maybe* accepts Medicare and value of ‘Y’ indicates a provider indeed accepts Medicare.

```r
# Reported Quality Measures
ggplot(provider,aes(x = Reported.Quality.Measures)) +
  geom_bar() +
  labs(title = "Frequency of providers reporting quality measures")
```



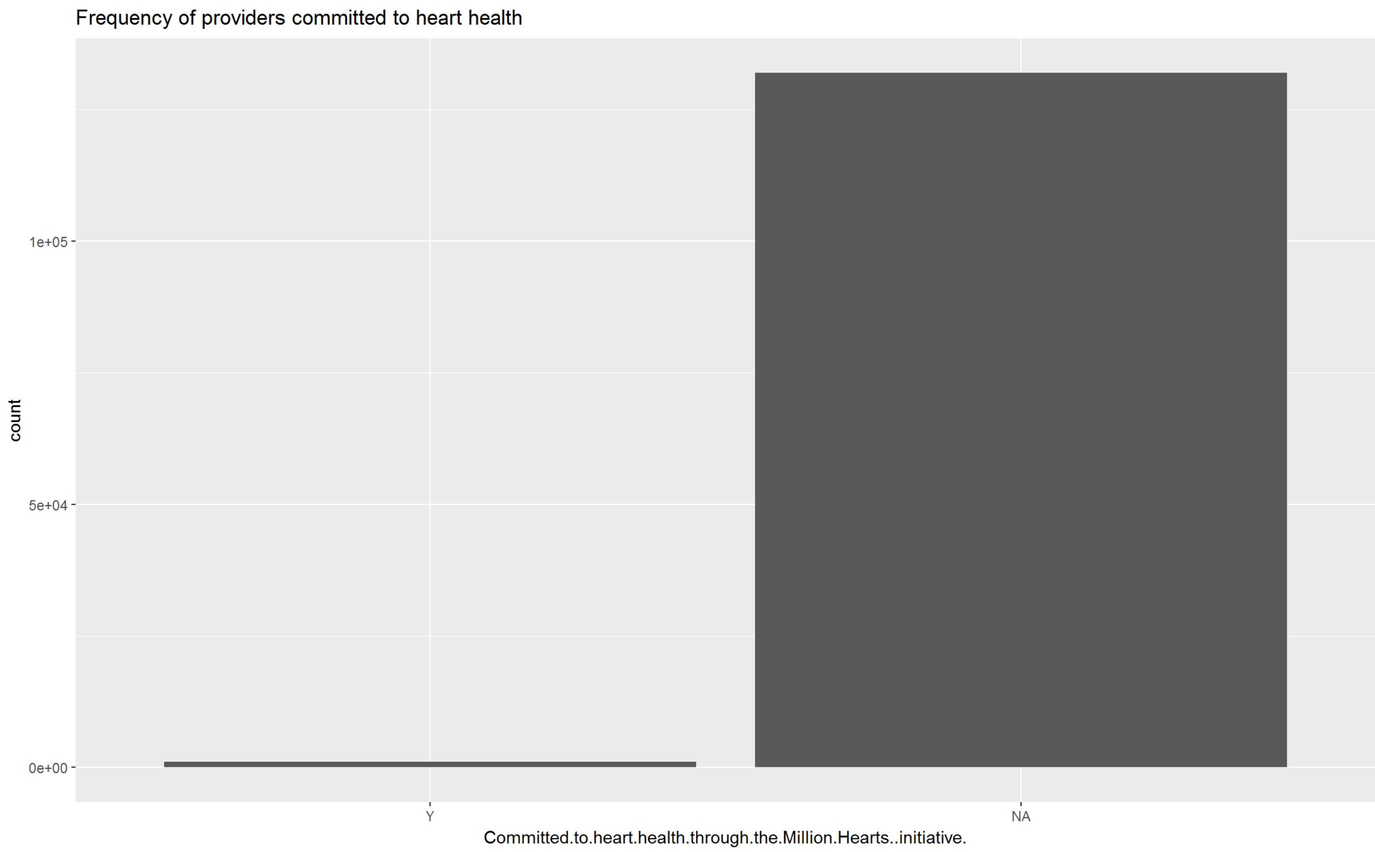
Unsurprised that this is the distribution of providers reporting quality measures because some NA values are expected here.

```r
# Used electronic health records
ggplot(provider,aes(x = Used.electronic.health.records)) +
  geom_bar() +
  labs(title = "Frequency of providers using electronic health records")
```



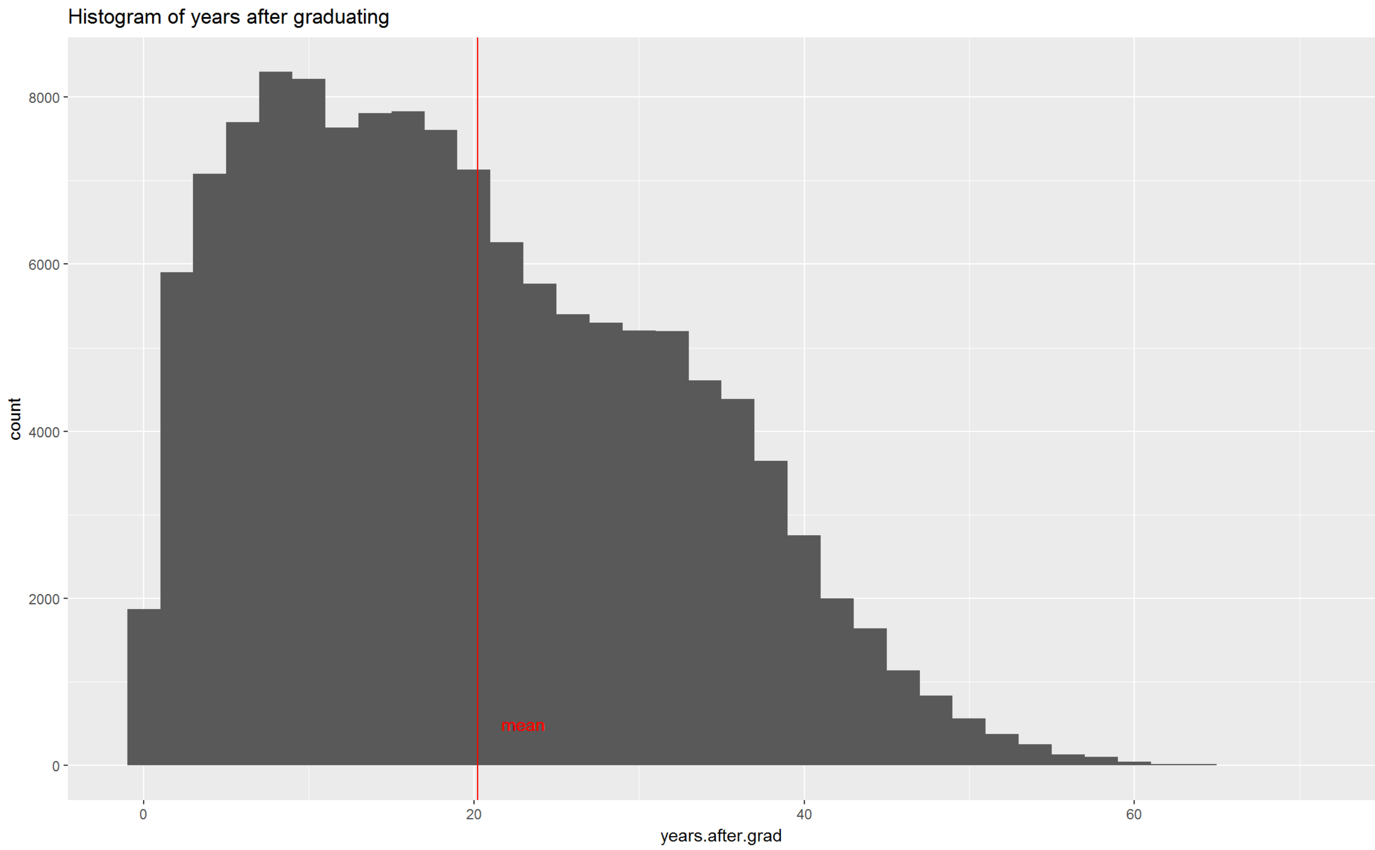
Did not expect so few providers to be using electronic health records! Since the only two values are ‘Y’ and NA, then the NA values could perhaps be ‘Y’ or some other value. However, instead of imputing a value, these were converted to ‘No Answer’ because there are no ‘N’ values to understand the distribution of actual values.

```
# Committed to heart health
ggplot(provider,aes(
  x = Committed.to.heart.health.through.the.Million.Hearts..initiative.)) +
  geom_bar() +
  labs(title = "Frequency of providers committed to heart health")
```



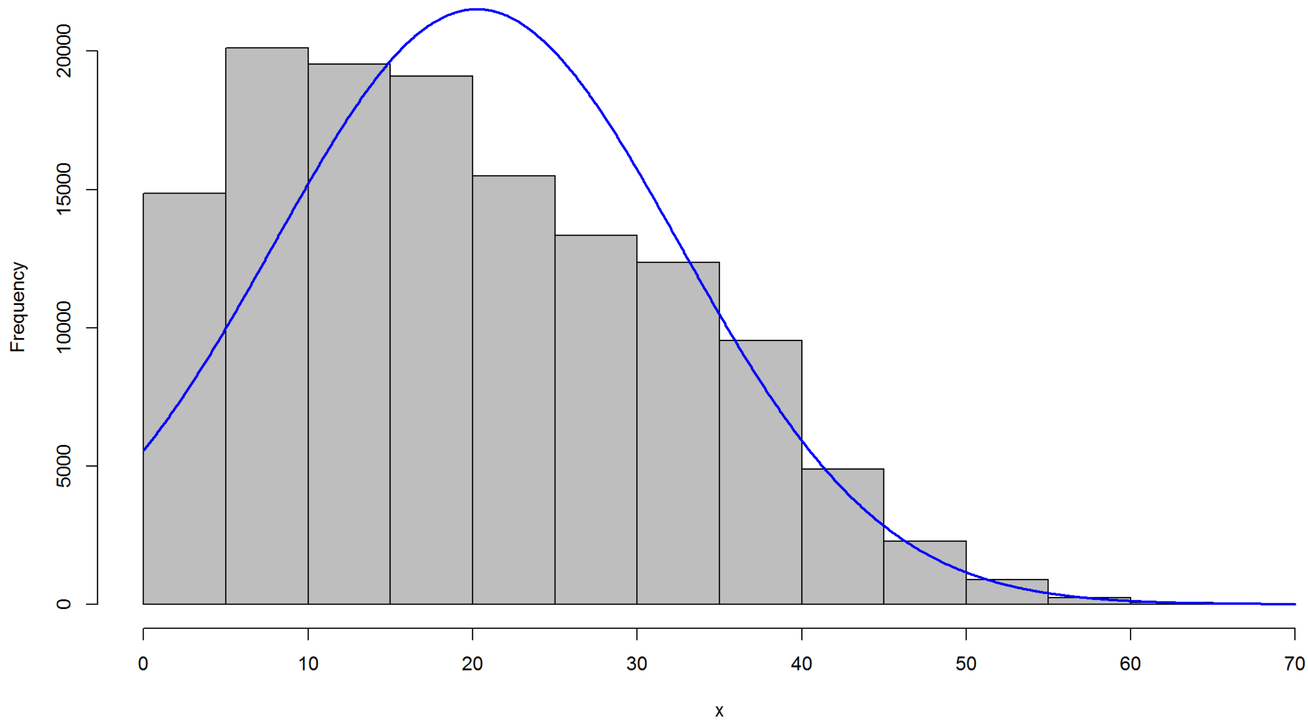
Surprised to see the high ratio of NA:‘Y’ values for this variable. Perhaps not many members utilize this offering

```r
# years after grad
provider %>%
  filter(!is.na(years.after.grad)) %>%
  ggplot(aes(x = years.after.grad)) +
  geom_histogram(binwidth = 2) +
  labs(title = "Histogram of years after graduating") +
  geom_vline(aes(xintercept = mean(years.after.grad)),colour = "red") +
  geom_text(aes(x = 23, label = "mean", y = 500), colour = "red")
```

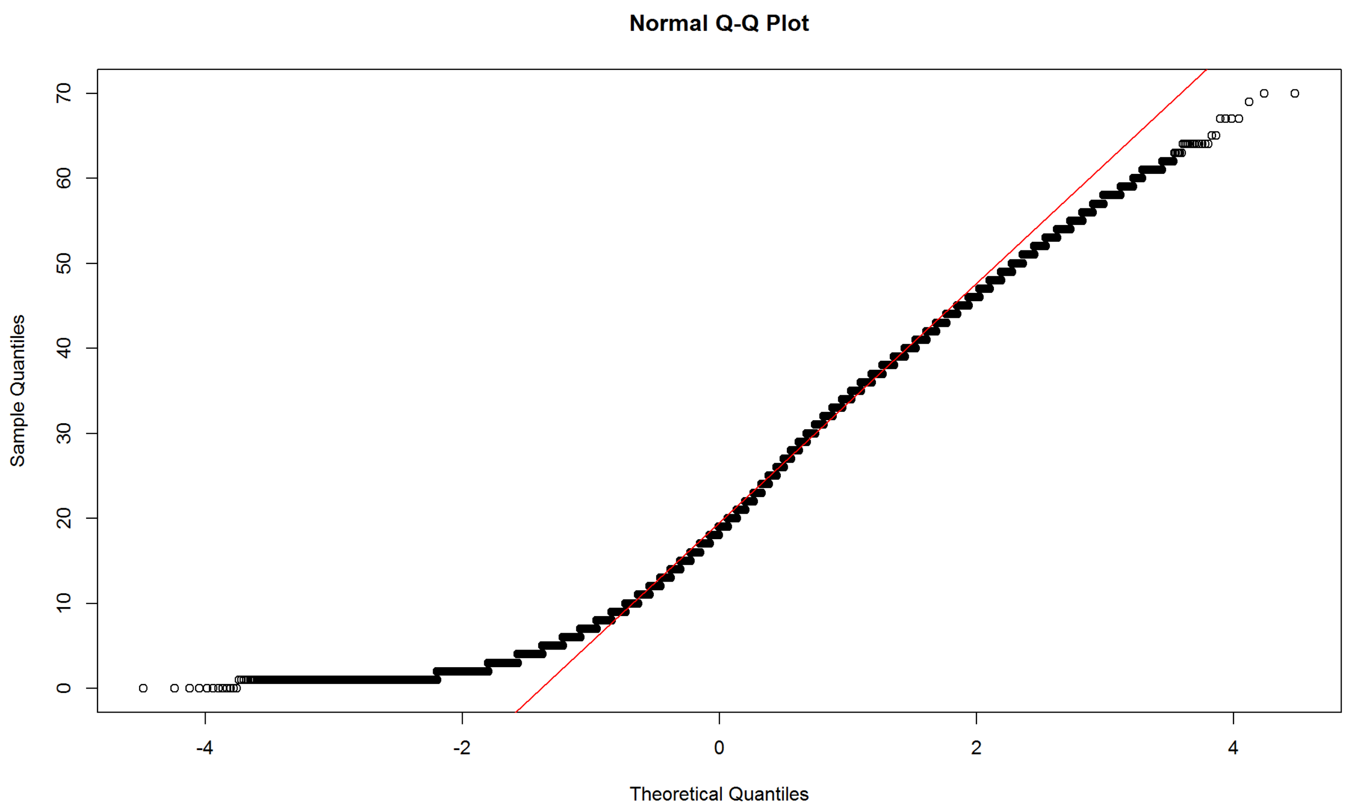


Do we need to normalize this data? After checking, possibly not.

```r
plotNormalHistogram(provider$years.after.grad)
```




```
qqnorm(provider$years.after.grad)
qqline(provider$years.after.grad, col = "red")
```



Hypothesis testing

In the above, we found that there were many providers with `NA` credentials. We'd like to know if there is a relationship between having `NA` credentials and various other dependent variables. Only some variables were checked to understand if PCA would help reduce correlation between variables.

```
# Create vector of yes/no NA credentials
provider <-
  provider %>%
  mutate(credential.isNA =
           factor(ifelse(is.na(Credential), 'Y', 'N')))

# Gender
chisq.test(table(provider$Gender,provider$credential.isNA))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(provider$Gender, provider$credential.isNA)
## X-squared = 2912.7, df = 1, p-value < 2.2e-16
```

With a p-value < 0.05, reject the null hypothesis that states there is no relationship between these variables.

```
# Medical school name
# fisher.test(table(provider$Medical.school.name,provider$credential.isNA))
# [TODO] Figure out way to bin medical schools

# Graduation year
# fisher.test(table(provider$Graduation.year,provider$credential.isNA))

# Primary specialty

# Group practice members

# State

# Medicare
chisq.test(provider$Professional.accepts.Medicare.Assignment,
           provider$credential.isNA)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  provider$Professional.accepts.Medicare.Assignment and provider$credential.isNA
## X-squared = 2.9427, df = 1, p-value = 0.08627
```

With a p-value > 0.05, do not reject the null hypothesis that states there is no relationship between these variables.

```
# Quality Measures
provider$Reported.Quality.Measures <-
  factor(provider$Reported.Quality.Measures,
         levels = levels(addNA(provider$Reported.Quality.Measures)),
         labels = c(levels(provider$Reported.Quality.Measures), "No Answer"),
         exclude=NULL)

chisq.test(provider$Reported.Quality.Measures,provider$credential.isNA)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  provider$Reported.Quality.Measures and provider$credential.isNA
## X-squared = 1117.2, df = 1, p-value < 2.2e-16
```

With a p-value < 0.05, reject the null hypothesis that states there is no relationship between these variables.

```
prop.table(xtabs(~Reported.Quality.Measures + credential.isNA, data = provider))
```

```
##                            credential.isNA
## Reported.Quality.Measures          N          Y
##                 Y         0.25669430 0.44490873
##                 No Answer 0.08091026 0.21748672
```

Higher proportion of providers have a credential and report quality measures.

```r
# electronic Health records
provider$Used.electronic.health.records <- factor(provider$Used.electronic.health.records, levels = levels(addNA(provider$Used.electronic.health.records)), labels = c(levels(provider$Used.electronic.health.records), "No Answer"),exclude=NULL)

chisq.test(provider$Used.electronic.health.records,provider$credential.isNA)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  provider$Used.electronic.health.records and provider$credential.isNA
## X-squared = 1538.8, df = 1, p-value < 2.2e-16
```

With a p-value < 0.05, reject the null hypothesis that states there is no relationship between these variables.

```r
prop.table(xtabs(~Used.electronic.health.records + credential.isNA, data = provider))
```

```
##                              credential.isNA
## Used.electronic.health.records         N         Y
##                      Y         0.1103103 0.1502995
##                      No Answer 0.2272942 0.5120960
```

```r
# heart health
provider$Committed.to.heart.health.through.the.Million.Hearts..initiative. <- factor(provider$Committed.to.heart.health.through.the.Million.Hearts..initiative., levels = levels(addNA(provider$Committed.to.heart.health.through.the.Million.Hearts..initiative.)), labels = c(levels(provider$Committed.to.heart.health.through.the.Million.Hearts..initiative.), "No Answer"),exclude=NULL)

chisq.test(provider$Committed.to.heart.health.through.the.Million.Hearts..initiative.,provider$credential.isNA)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  provider$Committed.to.heart.health.through.the.Million.Hearts..initiative. and provider$credential.isNA
## X-squared = 21.353, df = 1, p-value = 3.821e-06
```

With a p-value < 0.05, reject the null hypothesis that states there is no relationship between these variables.

```r
prop.table(xtabs(~Committed.to.heart.health.through.the.Million.Hearts..initiative. + credential.isNA, data = provider))
```

```
##                                                                  credential.isNA
## Committed.to.heart.health.through.the.Million.Hearts..initiative.           N
##                                                          Y         0.003359361
##                                                          No Answer 0.334245196
##                                                                  credential.isNA
## Committed.to.heart.health.through.the.Million.Hearts..initiative.           Y
##                                                          Y         0.004967646
##                                                          No Answer 0.657427796
```

```r
# secondary specialty
chisq.test(provider$has.secondary.specialty,provider$credential.isNA)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  provider$has.secondary.specialty and provider$credential.isNA
## X-squared = 761.27, df = 1, p-value < 2.2e-16
```

With a p-value < 0.05, reject the null hypothesis that states there is no relationship between these variables.

```r
prop.table(xtabs(~has.secondary.specialty + credential.isNA, data = provider))
```

```
##                          credential.isNA
## has.secondary.specialty          N          Y
##                         N 0.27649725 0.57968901
##                         Y 0.06110731 0.08270643
```

```
# hospital affiliation
chisq.test(provider$has.hospital.affiliation,provider$credential.isNA)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  provider$has.hospital.affiliation and provider$credential.isNA
## X-squared = 777.85, df = 1, p-value < 2.2e-16
```

```
prop.table(xtabs(~has.hospital.affiliation + credential.isNA, data = provider))
```

```
##                          credential.isNA
## has.hospital.affiliation          N          Y
##                         N 0.08199998 0.20957305
##                         Y 0.25560457 0.45282239
```

```
# remove column `credential.isNA` if moving on
col.remove <- list.append(col.remove,'credential.isNA')
```

Graph Cluster Analysis

I decided to use graph cluster analysis because I was interested in seeing how hospital networks helped contribute to provider clustering. Below are the pros and cons of using graph cluster analysis for this purpose:

Pros:
* Can analyze many-to-many relationships

Cons:
* Difficult to create the graph
* Difficult to analyze

Refer to this

For the graph cluster analysis portion, I opted to use only the first two columns of hospital associations because building an adjacency matrix was a bit more complicated than creating node and edge lists. However, using an adjacency matrix may have made this a more interesting and robust analysis.

Create node list

```
hosp1 <- provider %>%
  distinct(Hospital.affiliation.LBN.1) %>%
  dplyr::rename(label = Hospital.affiliation.LBN.1)

hosp2 <- provider %>%
  distinct(Hospital.affiliation.LBN.2) %>%
  dplyr::rename(label = Hospital.affiliation.LBN.2)

hosp3 <- provider %>%
  distinct(Hospital.affiliation.LBN.3) %>%
  dplyr::rename(label = Hospital.affiliation.LBN.3)

hosp4 <- provider %>%
  distinct(Hospital.affiliation.LBN.4) %>%
  dplyr::rename(label = Hospital.affiliation.LBN.4)

nodes <-
  join_all(list(hosp1,hosp2,hosp3,hosp4), by = "label", type = "full") %>%
  rowid_to_column("id")
```

id	label
1	NA

id	label
2	LITTLE RIVER HEALTHCARE
3	ST JOHN HOSPITAL AND MEDICAL CENTER
4	MEMORIAL HERMANN TEXAS MEDICAL CENTER
5	SSM HEALTH ST CLARE HOSPITAL - FENTON
6	HOAG ORTHOPEDIC INSTITUTE
7	ST ELIZABETH MEDICAL CENTER NORTH
8	ST JOHNS HOSPITAL
9	THOMAS JEFFERSON UNIVERSITY HOSPITAL
10	VALLEYCARE MEDICAL CENTER

Create edge list

```
# edges <- provider %>%
#   group_by_at(vars(Hospital.affiliation.LBN.1, Hospital.affiliation.LBN.2,
#                    Hospital.affiliation.LBN.3, Hospital.affiliation.LBN.4)) %>%
#   dplyr::summarise(weight = n()) %>%
#   ungroup()

edges <- provider %>%
  group_by_at(vars(Hospital.affiliation.LBN.1, Hospital.affiliation.LBN.2)) %>%
  dplyr::summarise(weight = n()) %>%
  ungroup()

edges$Hospital.affiliation.LBN.1 <- mapvalues(edges$Hospital.affiliation.LBN.1,
                                              from = nodes$label,
                                              to = nodes$id,
                                              warn_missing = FALSE)
edges$Hospital.affiliation.LBN.2 <- mapvalues(edges$Hospital.affiliation.LBN.2,
                                              from = nodes$label,
                                              to = nodes$id,
                                              warn_missing = FALSE)
# edges$Hospital.affiliation.LBN.3 <- mapvalues(edges$Hospital.affiliation.LBN.3,
#                                                from = nodes$label,
#                                                to = nodes$id)
# edges$Hospital.affiliation.LBN.4 <- mapvalues(edges$Hospital.affiliation.LBN.4,
#                                                from = nodes$label,
#                                                to = nodes$id)
```

Remove NA values from edge list.

```
edges <-
  edges %>%
  filter_all(all_vars(!is.na(.)))


edges$weight <- as.numeric(edges$weight)
```

Hospital.affiliation.LBN.1	Hospital.affiliation.LBN.2	weight
2170	4001	1
2170	1817	1
2050	231	1
2050	1146	1
2050	173	1
160	158	1
160	910	5
160	2010	7

Hospital.affiliation.LBN.1	Hospital.affiliation.LBN.2	weight
160	2814	1
160	3167	2

Remove values that have very small weights <10% of max weight.

```
edges <-
  edges %>%
  filter(weight >= ceiling(max(edges$weight) * .10))
nrow(edges)
```

```
## [1] 408
```

Remove the corresponding nodes that will not be used.

```
# rm.nodes <- union(union(union(edges$Hospital.affiliation.LBN.1,edges$Hospital.affiliation.LBN.2),edges$Hospital.affiliation.LBN.3), edges$Hospital.affiliation.LBN.4)

rm.nodes <- union(edges$Hospital.affiliation.LBN.1,edges$Hospital.affiliation.LBN.2)
```

Number of nodes to keep: 558.
Filter nodes list to keep nodes found in the first two hospital affiliation columns.

```
nodes <-
  nodes %>%
  filter(id %in% rm.nodes)
```

Visualize network

```
visNetwork(nodes,
           setNames(edges,c('from','to', 'value')),
           height = "700px", width = "100%") %>%
  visOptions(highlightNearest = TRUE,
             nodesIdSelection = list(style = 'width: 100%; height: 26px;
                                     background: #f8f8f8;
                                     color: darkblue;
                                     border:none;
                                     outline:none;')) %>%
  visPhysics(stabilization = FALSE) %>%
  visLayout(randomSeed = 100)
```

Select by id

Create network object for analysis purposes

```
hosp_tidy <- tbl_graph(nodes = nodes, edges = setNames(edges,c('from','to', 'value')), directed = FALSE)
hosp_tidy
```

```
## # A tbl_graph: 558 nodes and 408 edges
## #
## # An undirected multigraph with 223 components
## #
## # Node Data: 558 x 2 (active)
##      id label
##   <int> <fct>
## 1     3 ST JOHN HOSPITAL AND MEDICAL CENTER
## 2     4 MEMORIAL HERMANN TEXAS MEDICAL CENTER
## 3     7 ST ELIZABETH MEDICAL CENTER NORTH
## 4     8 ST JOHNS HOSPITAL
## 5    14 METHODIST HOSPITAL,THE
## 6    16 SPECTRUM HEALTH - BUTTERWORTH CAMPUS
## # ... with 552 more rows
## #
## # Edge Data: 408 x 3
##    from    to value
##   <int> <int> <dbl>
## 1     1     2    19
## 2     1     3    19
## 3     1     4    36
## # ... with 405 more rows
```

Fraction of edges present relative to total possible edges

```
edge_density(
  graph = hosp_tidy,
  loops = F
)
```

```
## [1] 0.002625432
```

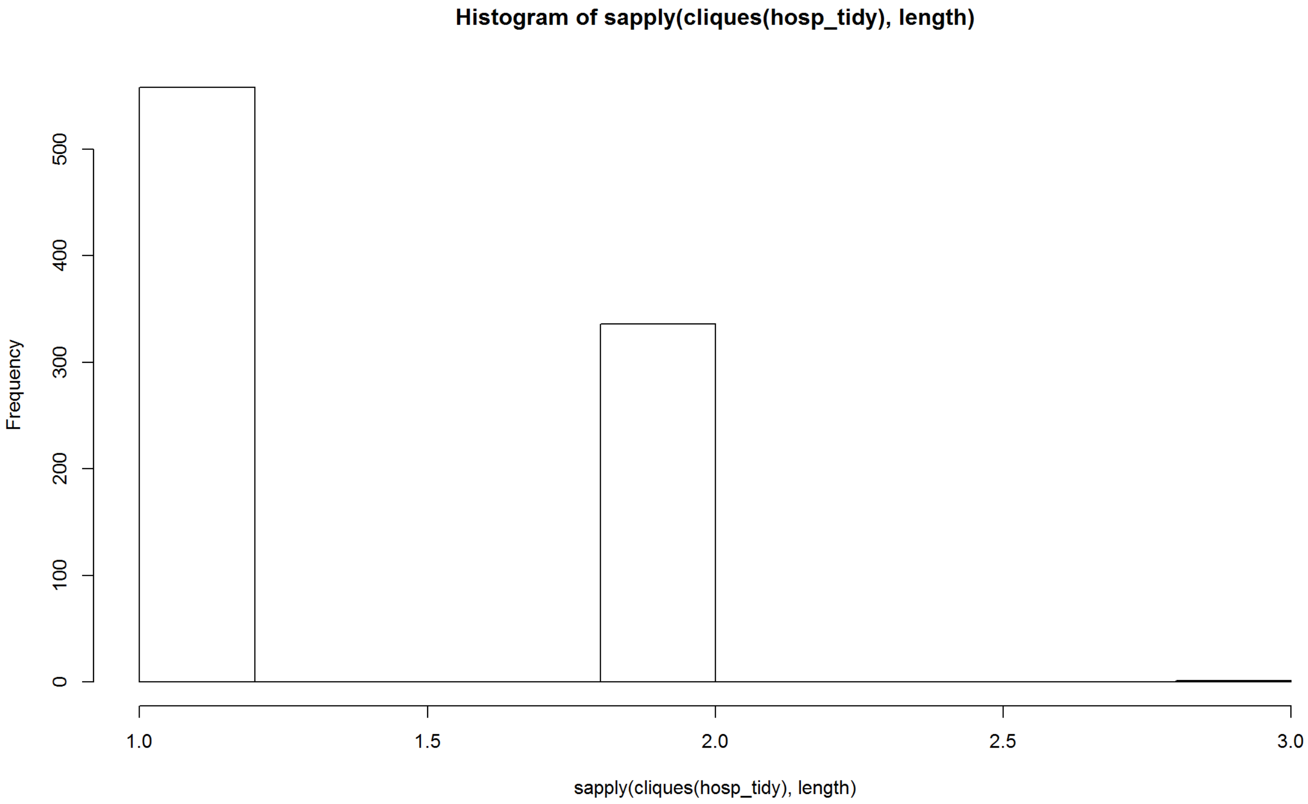
Fraction of triangles (completely connected 3 nodes) / all triangles

```
transitivity(
  graph = hosp_tidy,
  type = 'global' # 'local'
)
```

```
## [1] 0.01898734
```

With a transitivity = 1, the network contains all possible edges. In this case, not all edges are present between hospitals.
Find cliques and give clique sizes

```
hist(
  sapply(
    cliques(hosp_tidy),
    length
  )
) # clique sizes
```



Cliques are described as a set of nodes where all possible connections between nodes exist, which may indicate a stronger version of community. Source.
As shown here, we have a low number of cliques that form triangles (three node clique). As such, it may be better to look for network structures that are weaker to understand hospital associations.

Cliques with max number of nodes

```
largest_cliques(hosp_tidy)
```

```
## [[1]]
## + 3/558 vertices, from 6e8f8d0:
## [1] 425 422 423
```

Cluster using various method

The different clustering methods will be evaluated according to modularity and variation of information (VI). VI measures the amount of information lost and gained in changing from one clustering method to another method. In this evaluation, low VI indicates that the clusterings are fairly similar. Modularity measures how dense the connections are between nodes within modules. It looks to see if the number of edges in a cluster is comparable to the number of edges in a cluster found in a random network.

Refer to this article for info on VI.
Refer to this article for information on modularity.

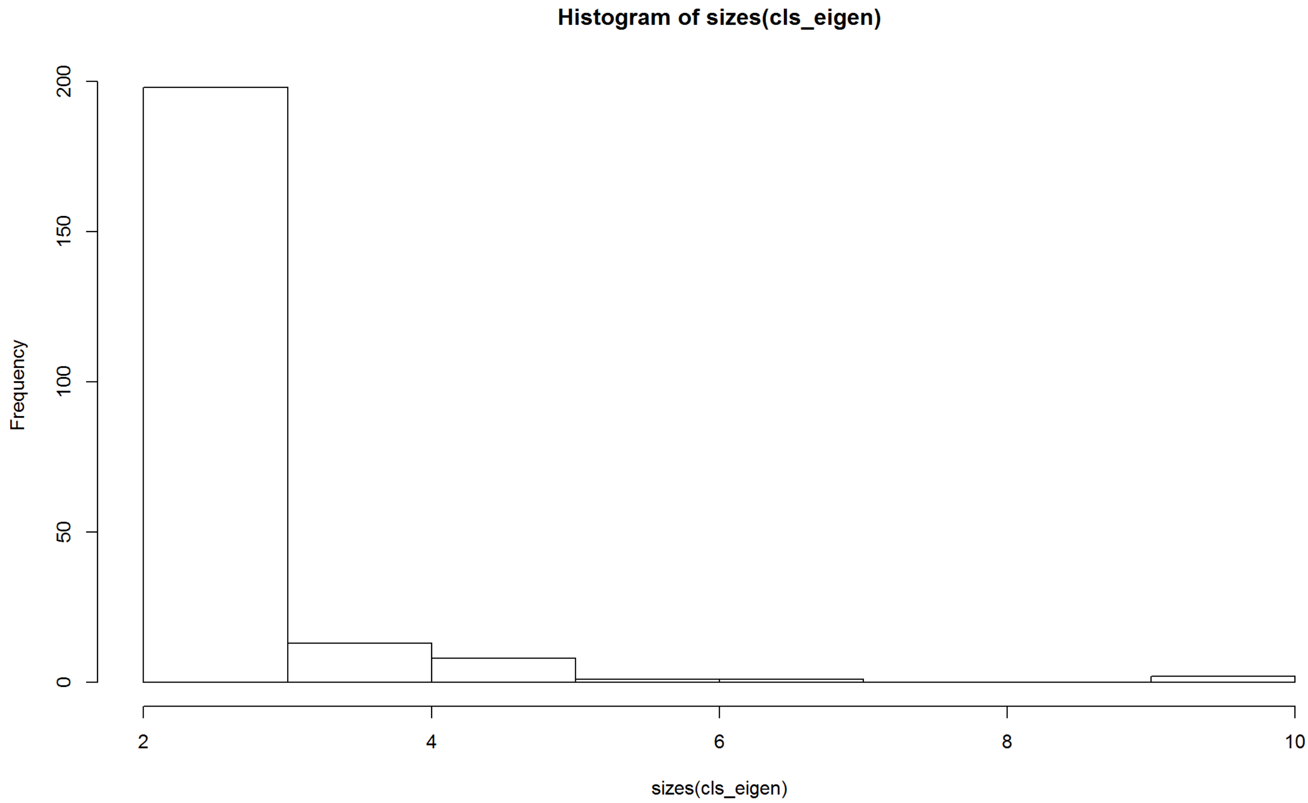
Get the leading Eigen value clusters
According to this document, leading eigenvector community structure is detected by finding “densely connected subgraphs by calculating the leading non-negative eigenvector of the modularity matrix of the graph.”

```
cls_eigen <-
  cluster_leading_eigen(hosp_tidy)

table(
  membership(cls_eigen)
)
```

```
##
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
##   7   2   2   2   2   2   3   2   2   2   4   2   2   2   2   4   2   4
##  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36
##   2   6   3   3   2   3  10   2   3   3   2   2   3   3   3   2   2   2
##  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54
##   3   2   3   5   5   2   2   2   2   3   2   2   2   2   2   2   3   2
##  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72
##   3   3   2   2   2   3   2   3   3   2   2   2   3   2   3   2   2   2
##  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90
##   3   2   2   2   4   2   2   2   3   2   4   4   2   5   5   4   2   2
##  91  92  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108
##   2   2   2   2   2   2   2   3   2   2   2   2   2   2   2   2   2   2
## 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126
##   2   2   3   2   3   2   2   2   2   2   2   2   3   2   2   2   2   2
## 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144
##   2   2   2   2  10   2   2   2   2   2   4   4   2   2   2   2   2   2
## 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162
##   2   2   2   2   3   2   2   2   2   2   2   3   2   4   2   3   2   2
## 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
##   2   2   2   2   2   2   2   3   4   2   3   5   2   2   2   2   2   2
## 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198
##   5   3   3   3   2   2   2   2   4   3   4   2   2   3   2   2   2   2
## 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216
##   2   2   5   2   2   2   2   2   2   2   2   2   2   2   2   2   5   2
## 217 218 219 220 221 222 223
##   2   2   2   3   2   2   2
```

Modularity of Leading Eigen community finding algorithm is 0.992635.

```
hist(sizes(cls_eigen))
```



Get the Louvain clusters

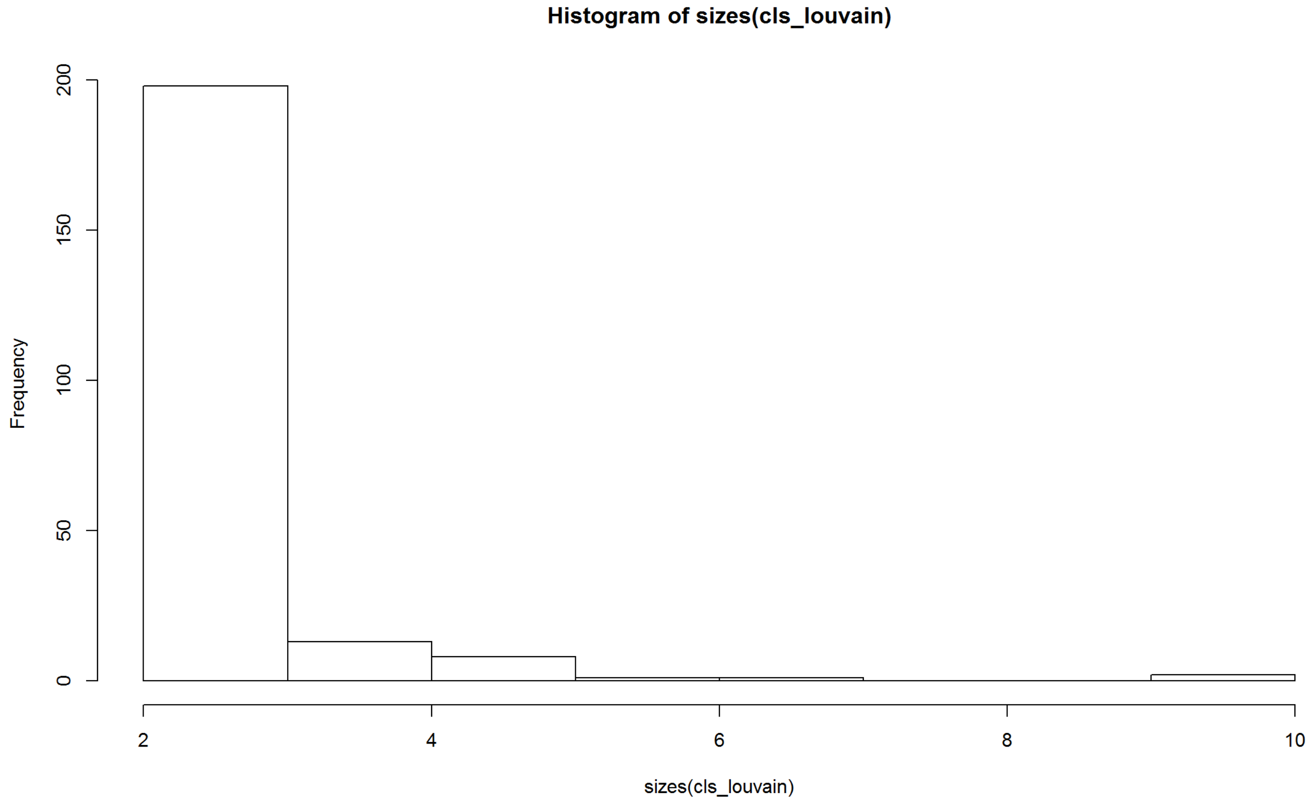
Using the Louvain function to find community structure implements both modularity optimization and a hierarchical approach. Source

```
cls_louvain <-
  cluster_louvain(hosp_tidy)

table(
  membership(cls_louvain)
)
```

```
##
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
##   2   2   2   2   2   3   2   2   2   4   2   2   2   2   4   2   4   2
##  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36
##   3   3   2   3   2   3   3   2   2   3   3   3   2   2   2   3   2   3
##  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54
##   2   2   2   2   3   6   2   2   2   2   2   2   3   2   3   5   3   2
##  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72
##   2   2   3   2   3   3   2   2   2   3   2   3   2   2   2   3   2   2
##  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90
##   2   4   2   2   2   3   2   4   2   5   5   4   2   2   2   2   2   2
##  91  92  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108
##   2   2   2   2   2   2   2   4   2   2   2   2   2   2   2   2   3   2
## 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126
##   3   2   2   2   2   2   2   2   3   7   2   2   2   2   2   2   2   2
## 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144
##   2   2   2   2   2   2   4   2   2   2   2   5   2   2   2   2   2   2
## 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162
##   4   3   2   2   2   2   2   2   3   2   4   2   3   2   2   2   2   2
## 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
##   2   2   2   2   3   4   2   3   5   2   2   2   2   2   2   5   3   3
## 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198
##   3   2   2  10   2   2   4   3   4   2   2   3   2   2   2   2  10   2
## 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216
##   2   5   2   2   2   2   2   2   2   2   2   2   2   2   2   5   2   2
## 217 218 219 220 221 222 223
##   3   2   2   3   2   2   2
```

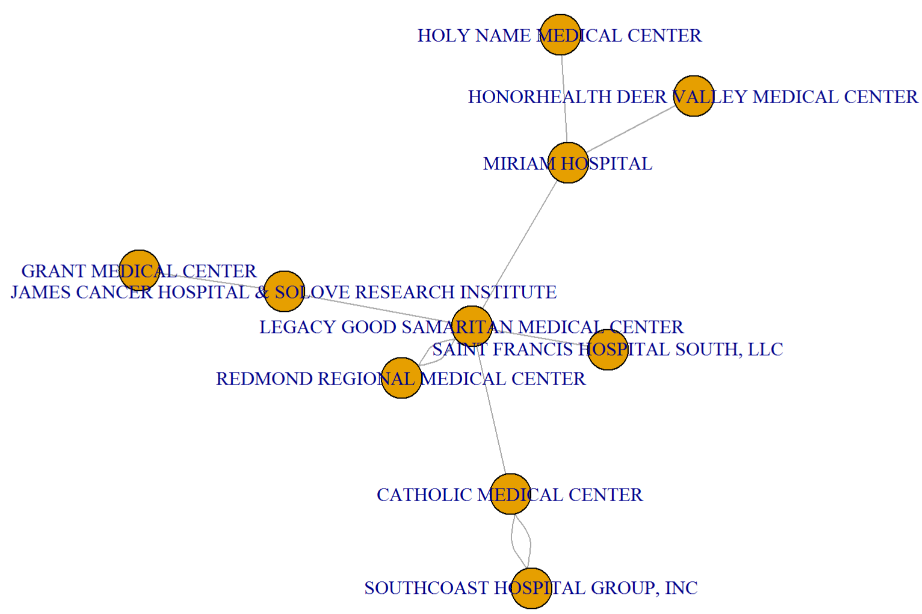
Modularity of Louvain community finding algorithm is 0.992635.

```
hist(sizes(cls_louvain))
```



Example of one of the largest communities in the hospital network

```
plot(induced_subgraph(hosp_tidy,cls_louvain[[184]]))
```



Get the Walktrap clusters

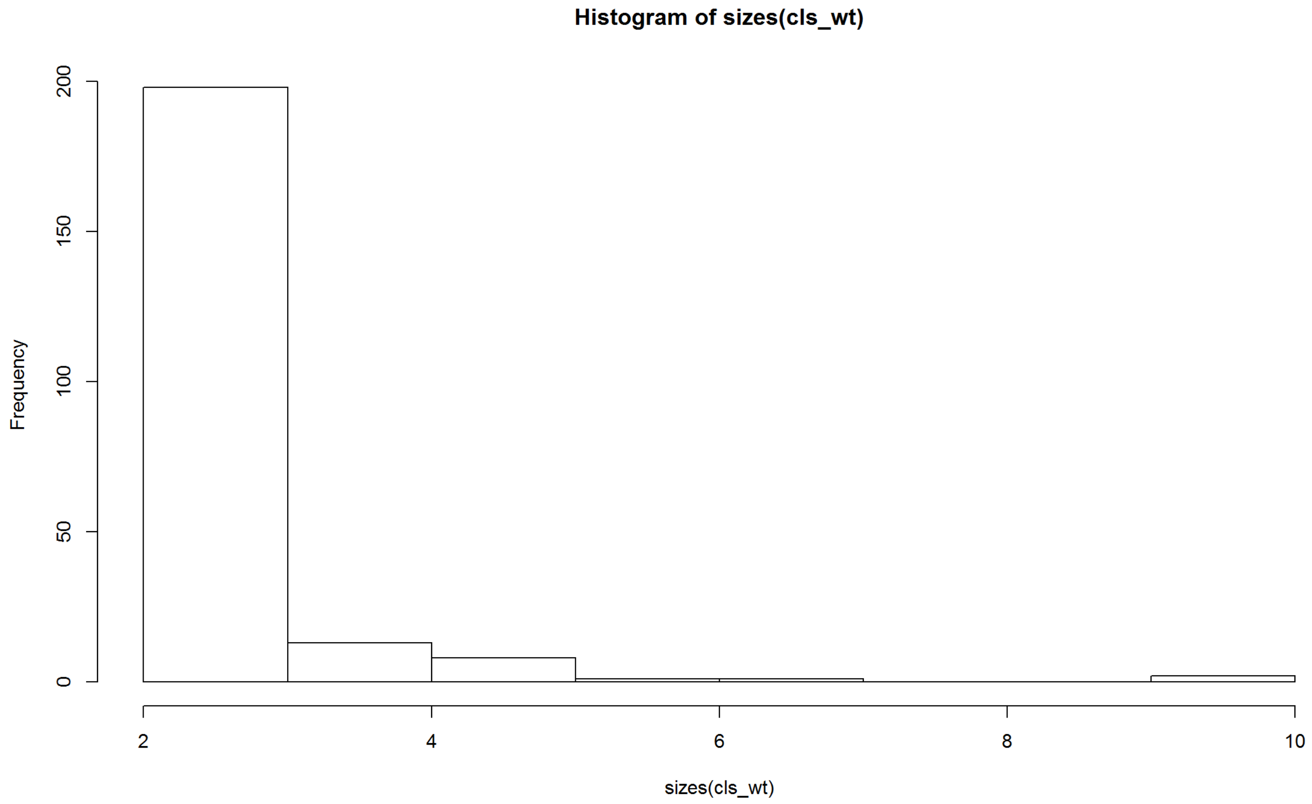
Community structure via short random walks is built on the idea that “short random walks tend to stay in the same community.” Source

```r
cls_wt <-
  cluster_walktrap(
    hosp_tidy,
    steps = 4
  )

table(
  membership(cls_wt)
)
```

```
##
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
##  10  10   7   6   5   5   5   4   4   5   4   4   4   5   4   4   4   4
##  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36
##   5   5   5   3   3   3   3   3   3   3   3   3   3   3   3   3   3   3
##  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54
##   3   3   3   3   4   4   4   4   3   3   3   3   3   3   3   3   3   3
##  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72
##   3   3   3   3   3   3   3   3   2   2   2   2   2   2   2   2   2   2
##  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90
##   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
##  91  92  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108
##   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
## 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126
##   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
## 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144
##   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
## 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162
##   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
## 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
##   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
## 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198
##   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
## 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216
##   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
## 217 218 219 220 221 222 223
##   2   2   2   2   2   2   2
```

Modularity of Walktrap community finding algorithm is 0.9926347.

```r
hist(sizes(cls_wt))
```



Modularity between the three clustering methods are the same and is fairly high, which indicates that the network has “dense connections between nodes within modules but sparse connections between nodes in different modules.” Source

In regard to hospital networks, this may be an accurate analysis because providers in a hospital network may be limited to a certain distance range. Therefore, providers within a certain area may be associated with the hospitals in that region, and distance is the deciding feature.

```
# compare(
#   cls_louvain,
#   cls_wt,
#   method = 'vi'
# )

# [TODO] Need to change edge list to have 2 columns with from and to.
# [UPDATE] Create adjacency matrix to encompass all of the hospital affiliations columns

# [TODO] Need to figure out a way to remove the NA
# [UPDATE] Removed NA from two column "from" "to" column
# [DONE] Removed values that had NA from edge list

# [TODO] Need to add weights.
# [UPDATE] Changed column name from "weights" to "value" which is what visNetwork wants
# [DONE]

# [TODO] How to handle providers that do not have a hospital affiliation? Maybe just calculate ratio of prov
iders without hospital affiliation and those with hospital affiliation?
# [DONE] Created new feature `has.hospital.affiliation`
```

Preprocess data

```
# Decide what to do with all NA
provider$Credential <-
  factor(provider$Credential,
         levels = levels(addNA(provider$Credential)),
         labels = c(levels(provider$Credential), "No Answer"),
         exclude = NULL)

provider$Medical.school.name <-
  factor(provider$Medical.school.name,
         levels = levels(addNA(provider$Medical.school.name)),
         labels = c(levels(provider$Medical.school.name), "No Answer"),
         exclude = NULL)
```

Impute NA values in numerical columns using mean since the variables have been transformed to a normal distribution.

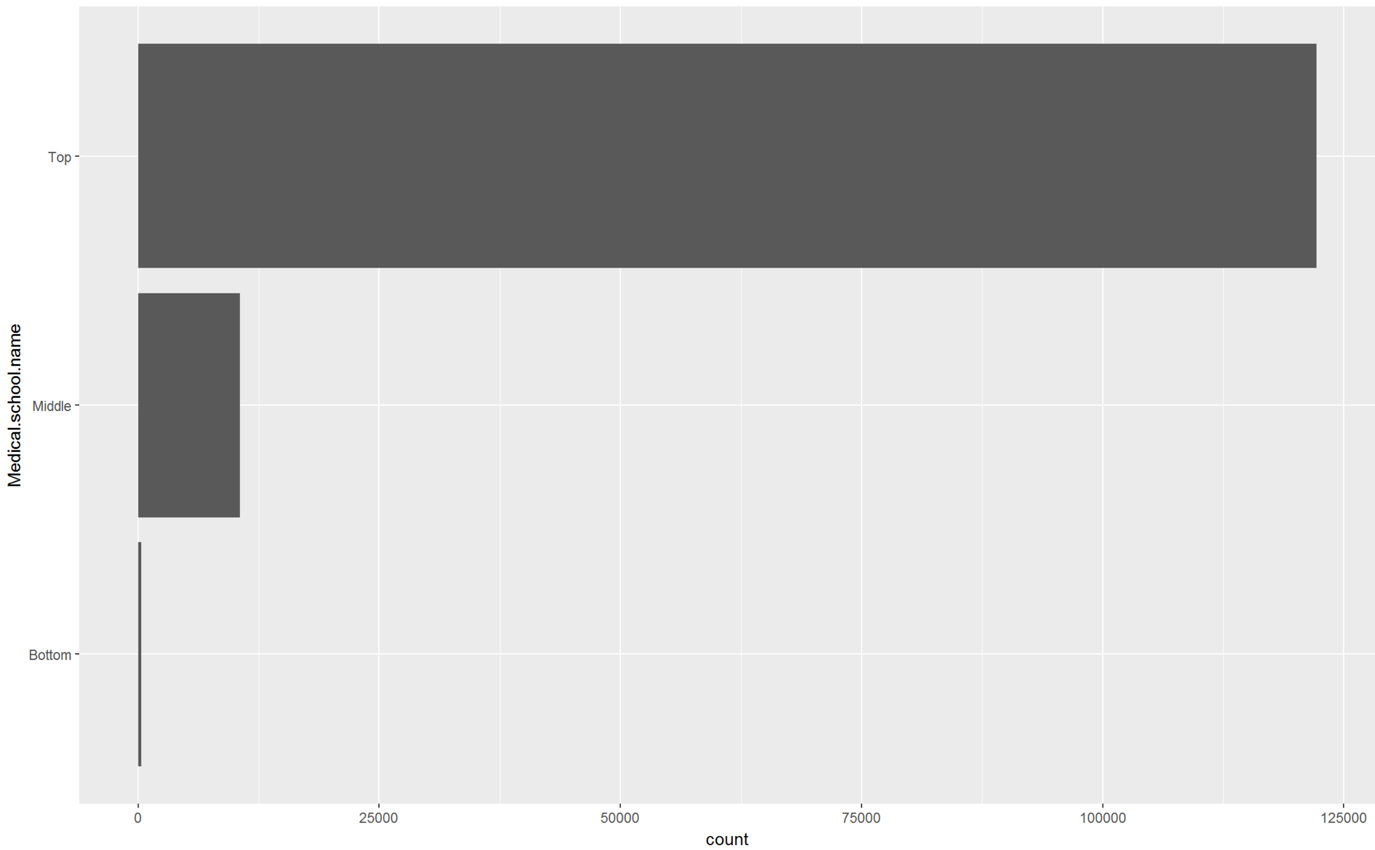
```
provider[is.na(provider[,"years.after.grad"]), "years.after.grad"] <-
  round(mean(provider$years.after.grad, na.rm = TRUE))
provider[is.na(provider[,"Number.of.Group.Practice.members_log"]),
         "Number.of.Group.Practice.members_log"] <-
  round(mean(provider$Number.of.Group.Practice.members_log, na.rm = TRUE))
```

Standardize and center variables from 0-1

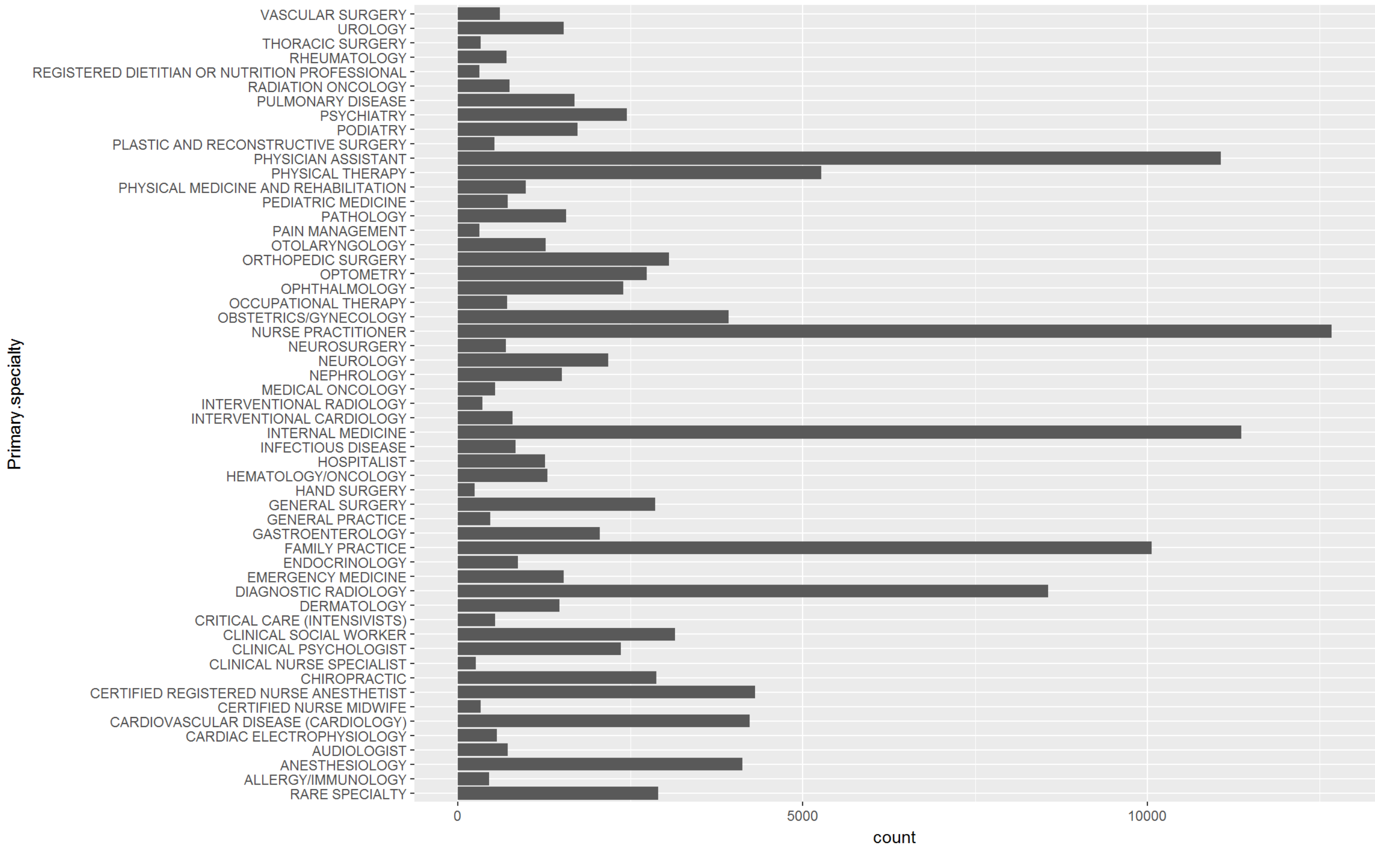
```
scale.center <- function(data) {
  output = (data - min(data)) / (max(data) - min(data))
  return(output)
}

provider$years.after.grad <- scale.center(provider$years.after.grad)

provider$Number.of.Group.Practice.members_log <- scale.center(provider$Number.of.Group.Practice.members_log)
```

Bin categorical variables that have too many factors

```
provider$Medical.school.name <-
  as.factor(case_when(provider$Medical.school.name %in% topthird.medical.school[[1]] ~ "Top",
                      provider$Medical.school.name %in% tailthird.medical.school[[1]] ~ "Bottom",
                      TRUE ~ "Middle"))


provider %>%
  ggplot(aes(x = Medical.school.name)) +
  geom_bar() +
  coord_flip()
```




```
levels(provider$Primary.specialty)[
  which(levels(provider$Primary.specialty) %in% tailthird.specialty[[1]])] <-
  "RARE SPECIALTY"

provider %>%
  ggplot(aes(x = Primary.specialty)) +
  geom_bar() +
  coord_flip()
```



Decide which columns to analyze

```r
identity.col <- c("NPI","PAC.ID","Professional.Enrollment.ID","Last.Name","First.Name","Middle.Name",
                  "Suffix","Organization.legal.name","Group.Practice.PAC.ID","Line.1.Street.Address",
                  "Line.2.Street.Address", "Marker.of.address.line.2.suppression","City","Zip.Code",
                  "Phone.Number")

cat("Number of columns left for analysis: ",
    ncol(provider) - (length(identity.col) + length(col.remove)))
```

```
## Number of columns left for analysis:  13
```

```r
processed.provider <- provider[ , !(names(provider) %in%
                                      list.append(identity.col,col.remove))]

summary(processed.provider)
```

```
##  Gender        Credential    Medical.school.name
##  F:57430   No Answer:88139   Bottom:   304
##  M:75631   MD       :31562   Middle: 10580
##            PA       : 2766   Top   :122177
##            NP       : 2023
##            DO       : 1866
##            CNA      : 1172
##            (Other)  : 5533
##             Primary.specialty     State
##  NURSE PRACTITIONER  :12674   CA     :10754
##  INTERNAL MEDICINE   :11366   TX     : 9452
##  PHYSICIAN ASSISTANT :11067   NY     : 8532
##  FAMILY PRACTICE     :10057   PA     : 7891
##  DIAGNOSTIC RADIOLOGY: 8560   FL     : 7452
##  PHYSICAL THERAPY    : 5271   MI     : 5805
##  (Other)             :74066   (Other):83175
##  Professional.accepts.Medicare.Assignment Reported.Quality.Measures
##  M:  4479                                 Y        :93356
##  Y:128582                                 No Answer:39705
##
##
##
##
##
##  Used.electronic.health.records
##  Y        :34677
##  No Answer:98384
##
##
##
##
##
##  Committed.to.heart.health.through.the.Million.Hearts..initiative.
##  Y        :  1108
##  No Answer:131953
##
##
##
##
##
##  years.after.grad has.secondary.specialty has.hospital.affiliation
##  Min.   :0.0000   N:113925                N:38797
##  1st Qu.:0.1429   Y: 19136                Y:94264
##  Median :0.2714
##  Mean   :0.2891
##  3rd Qu.:0.4143
##  Max.   :1.0000
##
##  Number.of.Group.Practice.members_log
##  Min.   :0.0000
##  1st Qu.:0.4306
##  Median :0.5325
##  Mean   :0.5503
##  3rd Qu.:0.7299
##  Max.   :1.0000
##
```

Complete one-hot encoding for the categorical variables

```
encoder <- onehot(processed.provider,max_levels = 350)

encode.processed.provider <- predict(encoder,processed.provider)

str(encode.processed.provider)
```

```
##  num [1:133061, 1:150] 1 1 0 1 1 0 0 1 0 1 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : NULL
##   ..$ : chr [1:150] "Gender=F" "Gender=M" "Credential=AA" "Credential=AU" ...
```

Principal Component Analysis

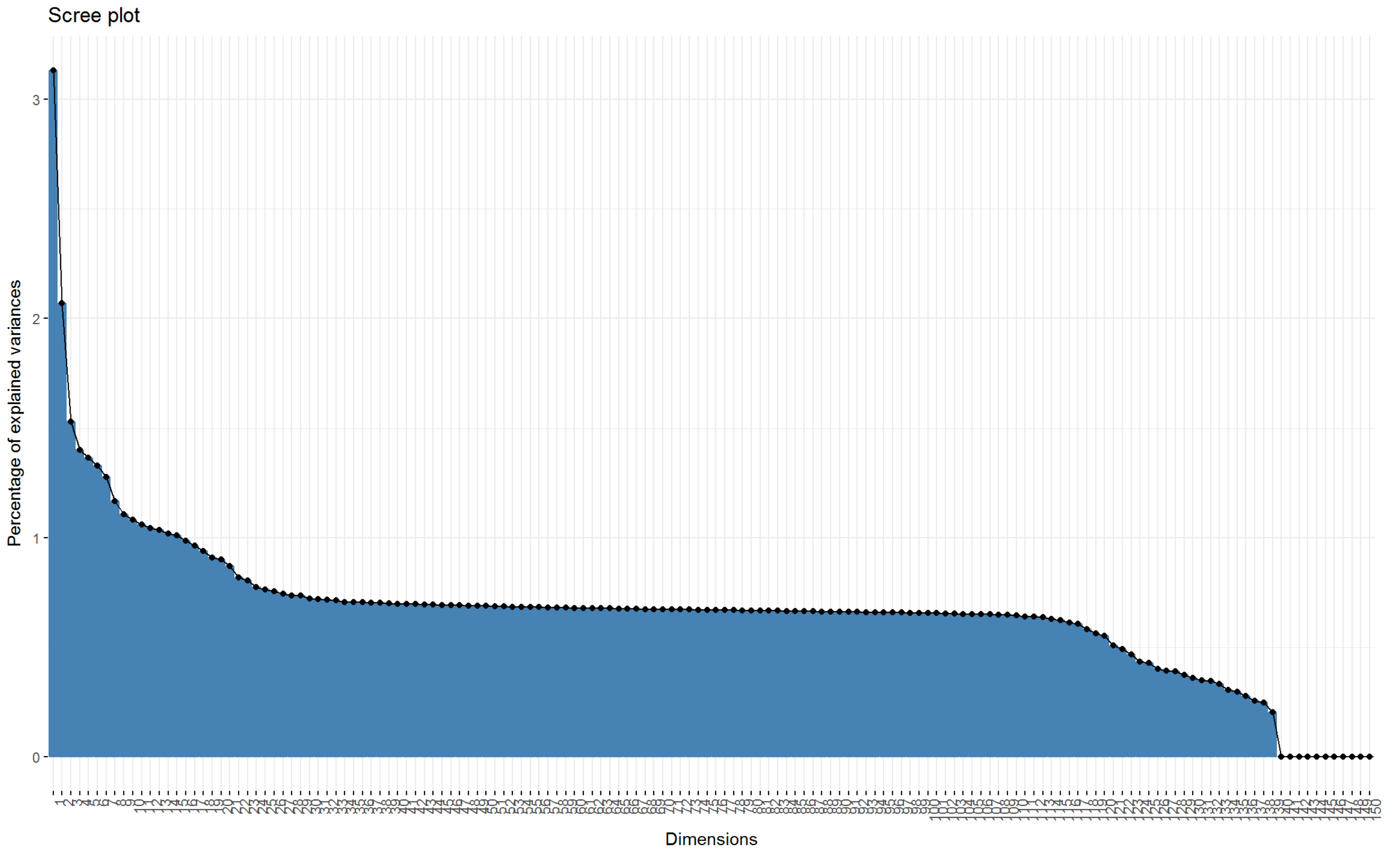
Principal component analysis was completed because there was evidence of endogeneity from the hypothesis testing results. Many subsequent machine learning algorithms require independent variables.

```
pca <- PCA(encode.processed.provider,graph = FALSE)
```

```
kable(head(pca$eig, n = 20))
```

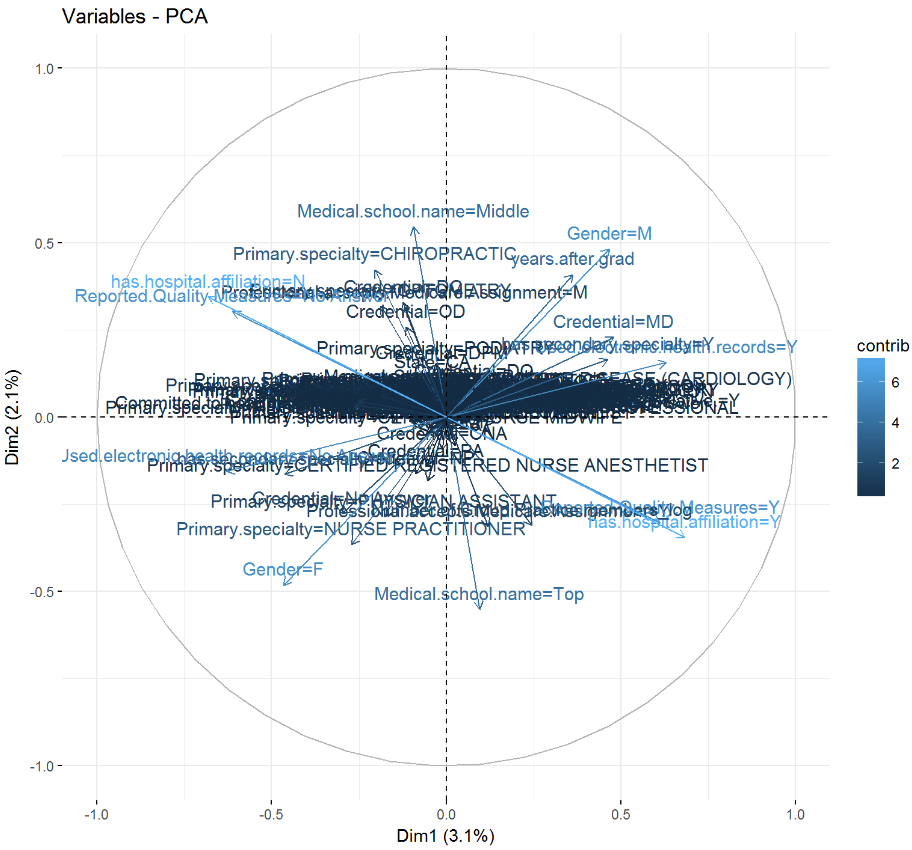
	eigenvalue	percentage of variance	cumulative percentage of variance
comp 1	4.699112	3.1327415	3.132741
comp 2	3.102769	2.0685127	5.201254
comp 3	2.291904	1.5279358	6.729190
comp 4	2.101767	1.4011778	8.130368
comp 5	2.047152	1.3647681	9.495136
comp 6	1.994164	1.3294429	10.824579
comp 7	1.913033	1.2753554	12.099934
comp 8	1.748977	1.1659850	13.265919
comp 9	1.659097	1.1060648	14.371984
comp 10	1.623727	1.0824848	15.454469
comp 11	1.588917	1.0592779	16.513747
comp 12	1.563208	1.0421389	17.555885
comp 13	1.551110	1.0340731	18.589959
comp 14	1.525877	1.0172517	19.607210
comp 15	1.513438	1.0089585	20.616169
comp 16	1.478635	0.9857567	21.601926
comp 17	1.443629	0.9624195	22.564345
comp 18	1.409683	0.9397886	23.504133
comp 19	1.362085	0.9080568	24.412190
comp 20	1.352184	0.9014563	25.313647

```
fviz_screeplot(pca, ncp = 150) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



Graph of variables

```
fviz_pca_var(pca, col.var = 'contrib')
```

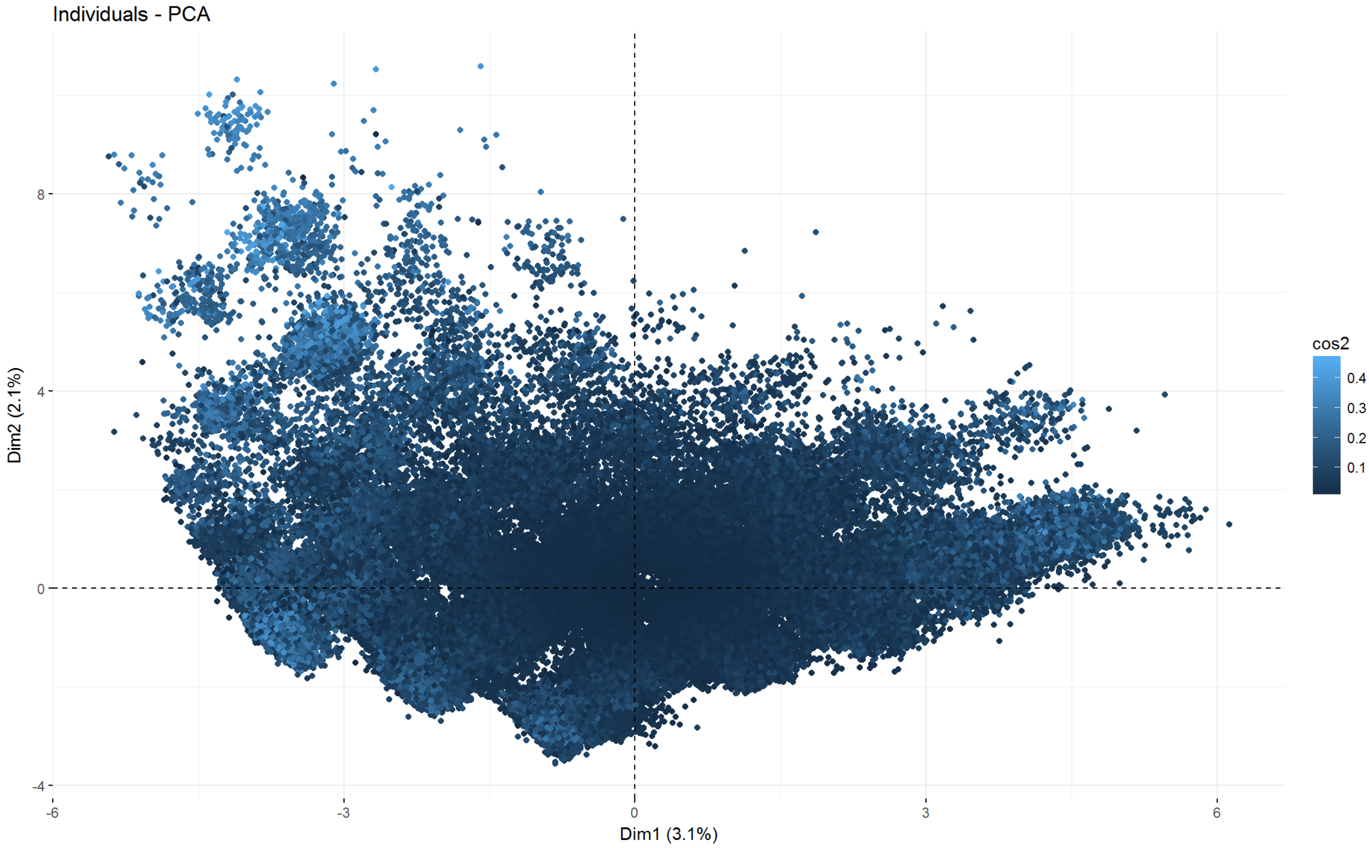


In this graph, you can understand variables that correlate positively and negatively. For example, some variables that correlate positively:

* Male gender + MD credential
* Reported quality measures + hospital affiliated
* Female gender + nurse practictioner primary specialty

Graph of individuals

```
fviz_pca_ind(pca, geom = 'point', col.ind = 'cos2')
```

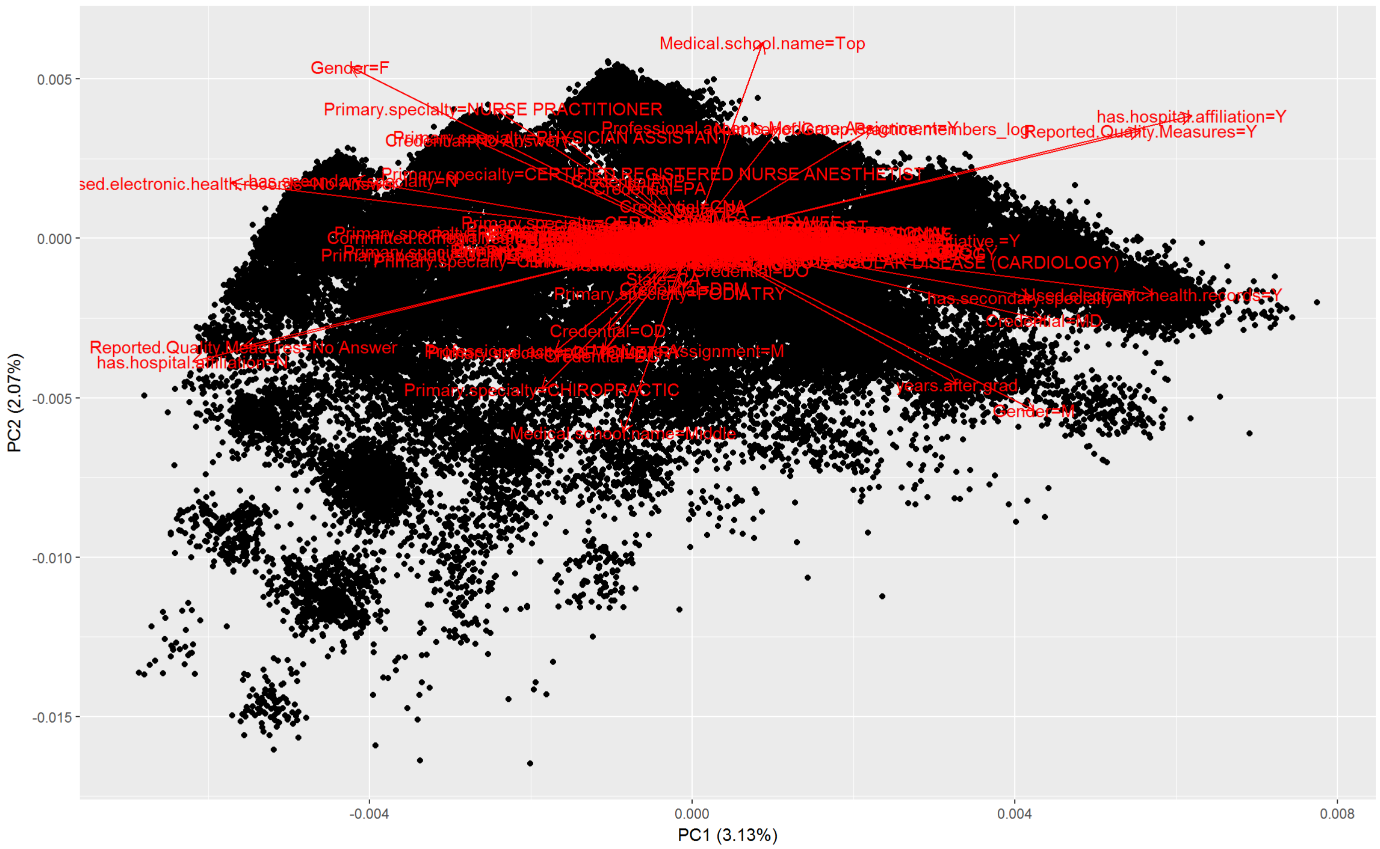


Overall, dimensionality reduction using PCA was not helpful because each principal component was responsible for a minute amount of variance. As a result, each variable contributed very little to each principal component. Looking at the scree plot, the elbow is actually closer to 125 principal components because each component explains only 3-1% of variance.

component	eigenvalue	percentage of variance	cumulative percentage of variance
comp 125	6.440340e-01	4.293560e-01	95.47559

125 component accounts for 95.5% of the variance in the data, so keeping 125 components reduces the number of predictors by 16.7%. This is another function that calculates PCA. For whatever reason, I am only able to get 5 dimensions from the original PCA results instead of the 125 components that would be necessary. This was not used for further analysis but for a quick sanity check. In future iterations, it may be more helpful to go through this route instead.

```
pca.prcomp <- prcomp(encode.processed.provider, scale. = TRUE)
autoplot(pca.prcomp,encode.processed.provider,
         loadings = TRUE,loadings.label = TRUE)
```




```
pca.provider <- pca.prcomp$x[,1:125]
```

K-Means

K-Means was chosen as a clustering method because of its simplicity. Below are the pros and cons of using this method.

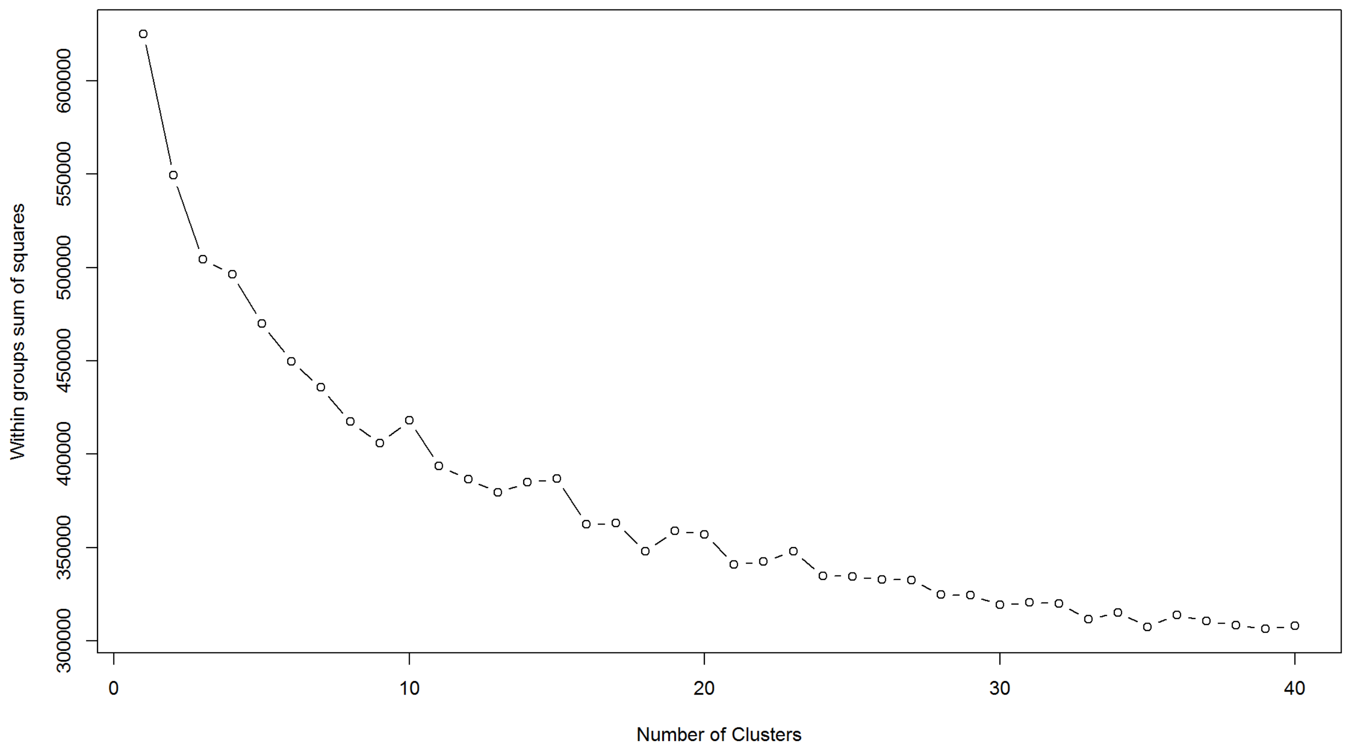
Pros:
* Fast to run

Cons:
* Only works well for spherical clusters
* Difficult to ascertain the number of clusters
* Difficult to work with outliers
Refer to this for more information

Determine number of clusters

```
wss <- (nrow(encode.processed.provider) - 1)*sum(apply(encode.processed.provider,2,var))
for (i in 2:40) wss[i] <- sum(kmeans(encode.processed.provider,
                                     centers = i,
                                     trace = TRUE)$withinss)
```

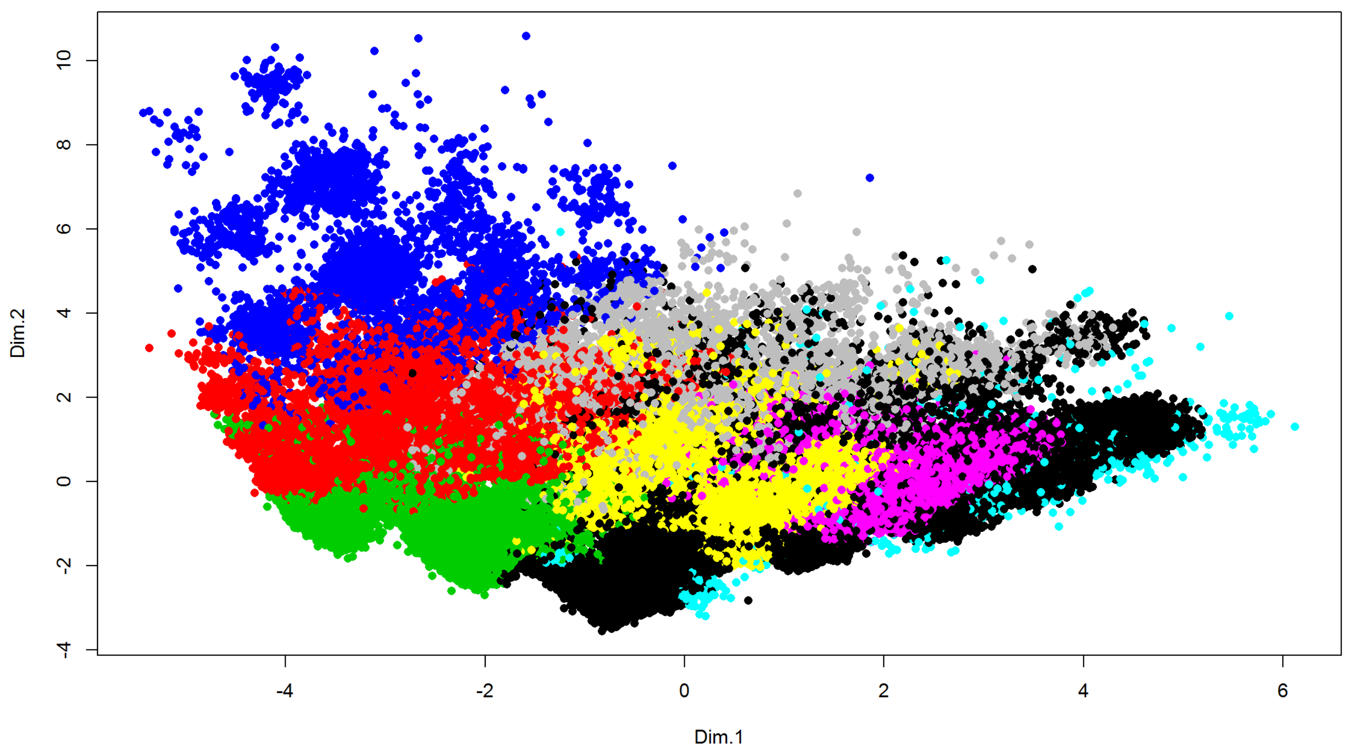
```
plot(1:40, wss, type = "b", xlab = "Number of Clusters",
     ylab = "Within groups sum of squares")
```



Refer to this document

For this scree plot, it is difficult to tell exactly where the elbow is since there is a slow gradient leveling off at 30 clusters. By visual inspection and acknowledging Occam's razor, 9 clusters was chosen as the elbow and model parameter.

```
k <- kmeans(x = pca$ind$coord, 9, , nstart=25, iter.max=1000)
plot(pca$ind$coord, col = k$clust,pch = 16)
```



As you can see, there is much overlap in clusters, which suggests that a Gaussian Mixture Model might be helpful because it allows for mixed cluster membership. Furthermore, this plot is only demonstrating principal component 1 and 2, which account for a low amount of variation in the data. Therefore, this representation must be taken with a grain of salt. In the future, it would be prudent to complete pairplots to analyze clusters across multiple dimensions.

```
# take 2
# k.prcomp <- kmeans(x = pca.provider, 9 , nstart = 25, iter.max = 1000)
# plot(pca.provider, col = k.prcomp$clust,pch=16)
```

How can we decipher these clusters?
For interpretation purposes, the most frequent value in each categorical variable and the mean value in each numerical variable is reported.

```
cluster.processed.provider <- cbind(processed.provider,factor(k$cluster))

for (cluster in sort(unique(k$cluster))) {
  temp <- cluster.processed.provider %>%
    filter(k$cluster == cluster)
  cat("Summarizing cluster: ", cluster)
  cat("\n")
  # summary.kmeans$`factor(k$cluster)`[iter] <- cluster
  for (col in names(temp)) {
    if (class(temp[[col]]) != "numerical") {
      print(paste(col, names(which.max(table(temp[[col]]))), sep = ": "))
    }
    else {
      print(paste(col, mean(temp[[col]]), sep =": "))
    }
  }
  cat("\n")
}
```

```
## Summarizing cluster:  1
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: INTERNAL MEDICINE"
## [1] "State: CA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.228571428571429"
## [1] "has.secondary.specialty: Y"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(k$cluster): 1"
##
## Summarizing cluster:  2
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: PHYSICAL THERAPY"
## [1] "State: CA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: No Answer"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.257142857142857"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: N"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(k$cluster): 2"
##
## Summarizing cluster:  3
## [1] "Gender: F"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: NURSE PRACTITIONER"
## [1] "State: TX"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: No Answer"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.0285714285714286"
## [1] "has.secondary.specialty: N"
```

```
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: N"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(k$cluster): 3"
##
## Summarizing cluster:  4
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Middle"
## [1] "Primary.specialty: CHIROPRACTIC"
## [1] "State: CA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: No Answer"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.257142857142857"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: N"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(k$cluster): 4"
##
## Summarizing cluster:  5
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: FAMILY PRACTICE"
## [1] "State: FL"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: Y"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: Y"
## [1] "years.after.grad: 0.285714285714286"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(k$cluster): 5"
##
## Summarizing cluster:  6
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: FAMILY PRACTICE"
## [1] "State: PA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: Y"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.285714285714286"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(k$cluster): 6"
##
## Summarizing cluster:  7
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: DIAGNOSTIC RADIOLOGY"
## [1] "State: CA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.285714285714286"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(k$cluster): 7"
##
## Summarizing cluster:  8
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Middle"
```

```
## [1] "Primary.specialty: FAMILY PRACTICE"
## [1] "State: OH"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.242857142857143"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(k$cluster): 8"
##
## Summarizing cluster:  9
## [1] "Gender: F"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: NURSE PRACTITIONER"
## [1] "State: PA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.114285714285714"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(k$cluster): 9"
```

K-Medoid (PAM)

K-Medoid was chosen as the next choice for clustering because K-Means worked relatively well. K-Medoid is implemented through the R function, partitioning around medoids (PAM), which “minimizes a sum of dissimilarities instead of a sum of squared euclidean distance.” Source

PAM works with medoids (samples of the dataset that represents the group) while K-Means works with centroids (artificially created entities that represent the cluster). As such, PAM could be more representative of the actual dataset.

In PAM, PCA is completed internally, so the encoded provider data was used instead of the PCA provider data. Furthermore, this data was sampled to fit into the memory allocation needed and to speed up run time.

```
sample.encode.processed.provider <-
  sample_n(as.data.frame(encode.processed.provider),
           size = ceiling(.1*nrow(encode.processed.provider)),
           replace = TRUE)
pamx <- pam(x = sample.encode.processed.provider, 9)
# summary(pamx)
```

How do these clusters differ?
To decipher, remove all columns from the medoid dataframe that are all 0’s. Print out the medoids.

```
pam.medoid <- as.data.frame(pamx$medoids)

pam.col.list = c()
for (col in names(pam.medoid)) {
  if (any(pam.medoid[[col]]) != 0) {
    print(col)
    pam.col.list <- list.append(pam.col.list,col)
  }
}
```

```
## [1] "Gender=F"
## [1] "Gender=M"
## [1] "Credential=MD"
## [1] "Credential=No Answer"
## [1] "Medical.school.name=Top"
## [1] "Primary.specialty=DIAGNOSTIC RADIOLOGY"
## [1] "Primary.specialty=FAMILY PRACTICE"
## [1] "Primary.specialty=INTERNAL MEDICINE"
## [1] "Primary.specialty=NURSE PRACTITIONER"
## [1] "Primary.specialty=PHYSICAL THERAPY"
## [1] "Primary.specialty=PHYSICIAN ASSISTANT"
## [1] "State=CA"
## [1] "State=FL"
## [1] "State=IL"
## [1] "State=NY"
## [1] "State=PA"
## [1] "State=TX"
## [1] "Professional.accepts.Medicare.Assignment=Y"
## [1] "Reported.Quality.Measures=Y"
## [1] "Reported.Quality.Measures=No Answer"
## [1] "Used.electronic.health.records=Y"
## [1] "Used.electronic.health.records=No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.=No Answer"
## [1] "years.after.grad"
## [1] "has.secondary.specialty=N"
## [1] "has.hospital.affiliation=N"
## [1] "has.hospital.affiliation=Y"
## [1] "Number.of.Group.Practice.members_log"
```

```
pam.medoid[pam.col.list]
```

```
##        Gender=F Gender=M Credential=MD Credential=No Answer
## 60319         0        1             0                    1
## 88693         0        1             0                    1
## 109463        0        1             1                    0
## 100123        0        1             1                    0
## 102081        0        1             0                    1
## 113620        0        1             0                    1
## 111678        1        0             0                    1
## 100867        1        0             0                    1
## 121706        1        0             0                    1
##        Medical.school.name=Top Primary.specialty=DIAGNOSTIC RADIOLOGY
## 60319                        1                                      0
## 88693                        1                                      0
## 109463                       1                                      0
## 100123                       1                                      1
## 102081                       1                                      0
## 113620                       1                                      1
## 111678                       1                                      0
## 100867                       1                                      0
## 121706                       1                                      0
##        Primary.specialty=FAMILY PRACTICE
## 60319                                  0
## 88693                                  1
## 109463                                 0
## 100123                                 0
## 102081                                 0
## 113620                                 0
## 111678                                 0
## 100867                                 0
## 121706                                 0
##        Primary.specialty=INTERNAL MEDICINE
## 60319                                    0
## 88693                                    0
## 109463                                   1
## 100123                                   0
## 102081                                   1
## 113620                                   0
## 111678                                   0
## 100867                                   0
## 121706                                   0
```

```
##        Primary.specialty=NURSE PRACTITIONER
## 60319                                  0
## 88693                                  0
## 109463                                 0
## 100123                                 0
## 102081                                 0
## 113620                                 0
## 111678                                 0
## 100867                                 1
## 121706                                 1
##        Primary.specialty=PHYSICAL THERAPY
## 60319                                1
## 88693                                0
## 109463                               0
## 100123                               0
## 102081                               0
## 113620                               0
## 111678                               0
## 100867                               0
## 121706                               0
##        Primary.specialty=PHYSICIAN ASSISTANT State=CA State=FL State=IL
## 60319                                   0        1        0        0
## 88693                                   0        0        1        0
## 109463                                  0        0        0        1
## 100123                                  0        0        0        0
## 102081                                  0        1        0        0
## 113620                                  0        0        0        0
## 111678                                  1        1        0        0
## 100867                                  0        0        1        0
## 121706                                  0        0        0        0
##        State=NY State=PA State=TX
## 60319         0        0        0
## 88693         0        0        0
## 109463        0        0        0
## 100123        1        0        0
## 102081        0        0        0
## 113620        0        0        1
## 111678        0        0        0
## 100867        0        0        0
## 121706        0        1        0
##        Professional.accepts.Medicare.Assignment=Y
## 60319                                        1
## 88693                                        1
## 109463                                       1
## 100123                                       1
## 102081                                       1
## 113620                                       1
## 111678                                       1
## 100867                                       1
## 121706                                       1
##        Reported.Quality.Measures=Y Reported.Quality.Measures=No Answer
## 60319                            0                                   1
## 88693                            1                                   0
## 109463                           1                                   0
## 100123                           1                                   0
## 102081                           0                                   1
## 113620                           1                                   0
## 111678                           0                                   1
## 100867                           1                                   0
## 121706                           1                                   0
##        Used.electronic.health.records=Y
## 60319                                0
## 88693                                1
## 109463                               1
## 100123                               0
## 102081                               0
## 113620                               0
## 111678                               0
## 100867                               0
## 121706                               0
##        Used.electronic.health.records=No Answer
## 60319                                        1
## 88693                                        0
```

```
## 88693                                          0
## 109463                                         0
## 100123                                         1
## 102081                                         1
## 113620                                         1
## 111678                                         1
## 100867                                         1
## 121706                                         1
##        Committed.to.heart.health.through.the.Million.Hearts..initiative.=No Answer
## 60319                                                                              1
## 88693                                                                              1
## 109463                                                                             1
## 100123                                                                             1
## 102081                                                                             1
## 113620                                                                             1
## 111678                                                                             1
## 100867                                                                             1
## 121706                                                                             1
##        years.after.grad has.secondary.specialty=N
## 60319          0.2714286                         1
## 88693          0.2857143                         1
## 109463         0.3857143                         1
## 100123         0.4571429                         1
## 102081         0.3428571                         1
## 113620         0.2285714                         1
## 111678         0.2000000                         1
## 100867         0.1000000                         1
## 121706         0.1571429                         1
##        has.hospital.affiliation=N has.hospital.affiliation=Y
## 60319                          1                          0
## 88693                          0                          1
## 109463                         0                          1
## 100123                         0                          1
## 102081                         0                          1
## 113620                         0                          1
## 111678                         1                          0
## 100867                         1                          0
## 121706                         0                          1
##        Number.of.Group.Practice.members_log
## 60319                             0.3348755
## 88693                             0.6130291
## 109463                            0.5325392
## 100123                            0.5325392
## 102081                            0.5325392
## 113620                            0.5899839
## 111678                            0.5325392
## 100867                            0.5394722
## 121706                            0.5999897
```

Agglomerative Hierarchical Clustering

Agglomerative hierarchical clustering was initially chosen because it is more informative than the flat unstructured clusters from K-Means and for its ease of implementation. However, some cons I experienced were that hierarchical clustering was not suitable for large datasets and since points assigned to a cluster cannot be moved around, order of the data and the initial seeds have a strong impact. Source

```
# d <- dist(pca$ind$coord, method = 'euclidean')
# hc1 <- hclust(d, method = 'complete')
# plot(hc1,cex=0.6,hang=-1)
#
# hc2 <- agnes(pca$ind$coord, method = 'complete')
```

Using the traditional hclust function with a distance matrix was unsuccessful due to the high computational complexity and memory allocation.

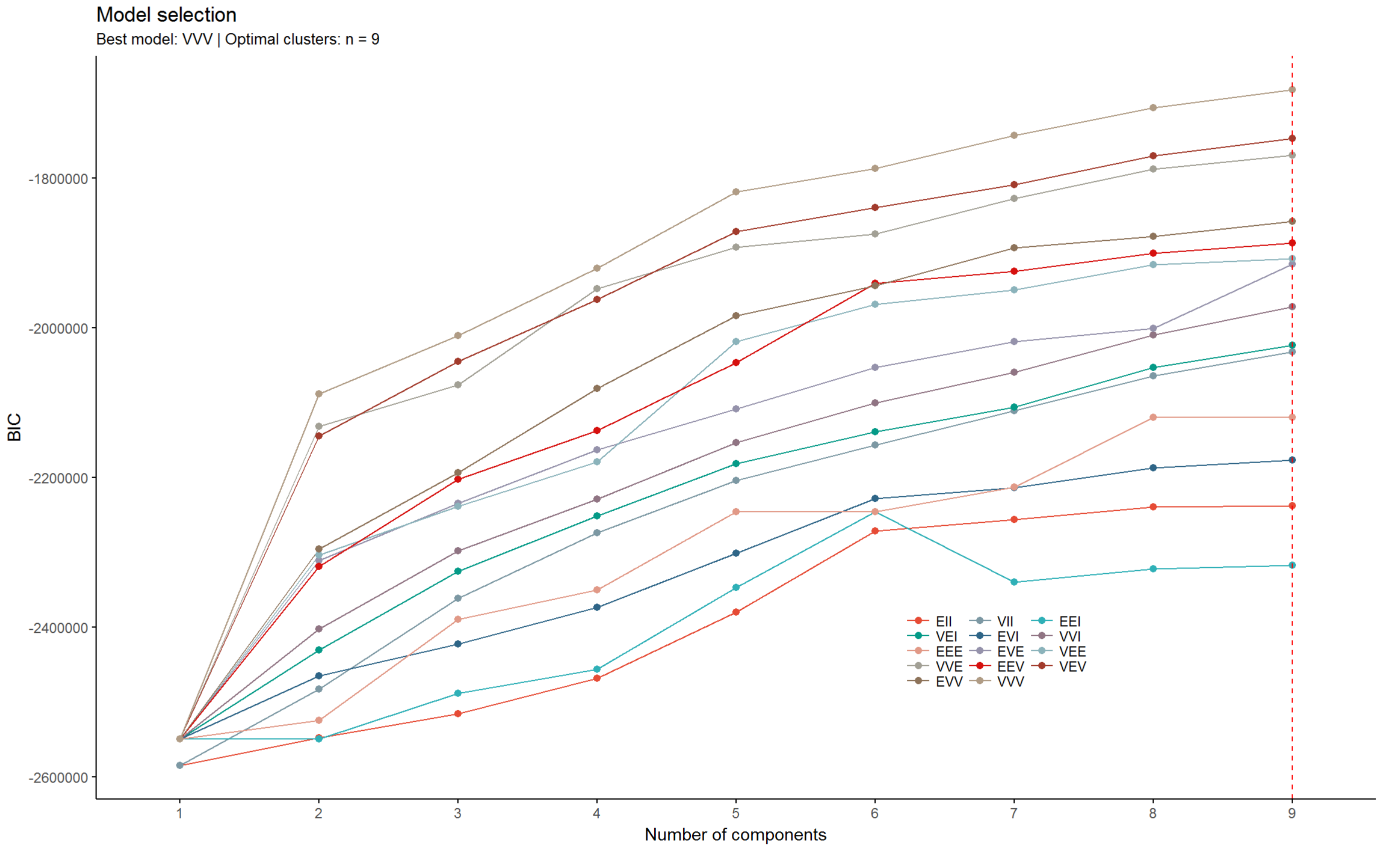
```
# Trying HCPC function which completes hierarchical clustering on Principle Components (NCPC)

# hc <- HCPC(pca.provider, nb.clust = -1)
```

In future work, it may be helpful to sample a smaller proportion of the data initially so that AHC may be used in this analysis.

Gaussian Mixture Model

Gaussian Mixture Model is a parametric model that assumes that the data points are generated from Gaussian distributions. Refer to this document. and this.

```
fit <- Mclust(pca$ind$coord)
fviz_mclust(
  fit,
  what = 'BIC',
  palette = 'npg'
)
```



VVV is the best fit with 9 clusters

```
summary(fit, parameters = TRUE)
```

```
## ----------------------------------------------------
## Gaussian finite mixture model fitted by EM algorithm
## ----------------------------------------------------
##
## Mclust VVV (ellipsoidal, varying volume, shape, and orientation) model with 9 components:
##
##  log.likelihood      n  df       BIC       ICL
##       -839817.8 133061 188 -1681854 -1696174
##
## Clustering table:
##     1     2     3     4     5     6     7     8     9
## 27287 22169 19422 16549  5570  8991 18113  9773  5187
##
## Mixing probabilities:
##          1          2          3          4          5          6
## 0.21320978 0.15955173 0.15081516 0.12348042 0.03988591 0.06621059
##          7          8          9
## 0.13471645 0.07423848 0.03789147
##
## Means:
##              [,1]       [,2]       [,3]       [,4]       [,5]       [,6]
## Dim.1 -0.25760612 -0.1649629 -0.7364784  2.5916190 -2.2232883 -3.3607102
## Dim.2 -0.34537883 -1.8464567  2.5844256  0.4369785 -1.9238568 -0.6434179
## Dim.3  0.17933025 -0.7407867 -1.5036058  2.0083183 -0.2571089  0.8413845
```

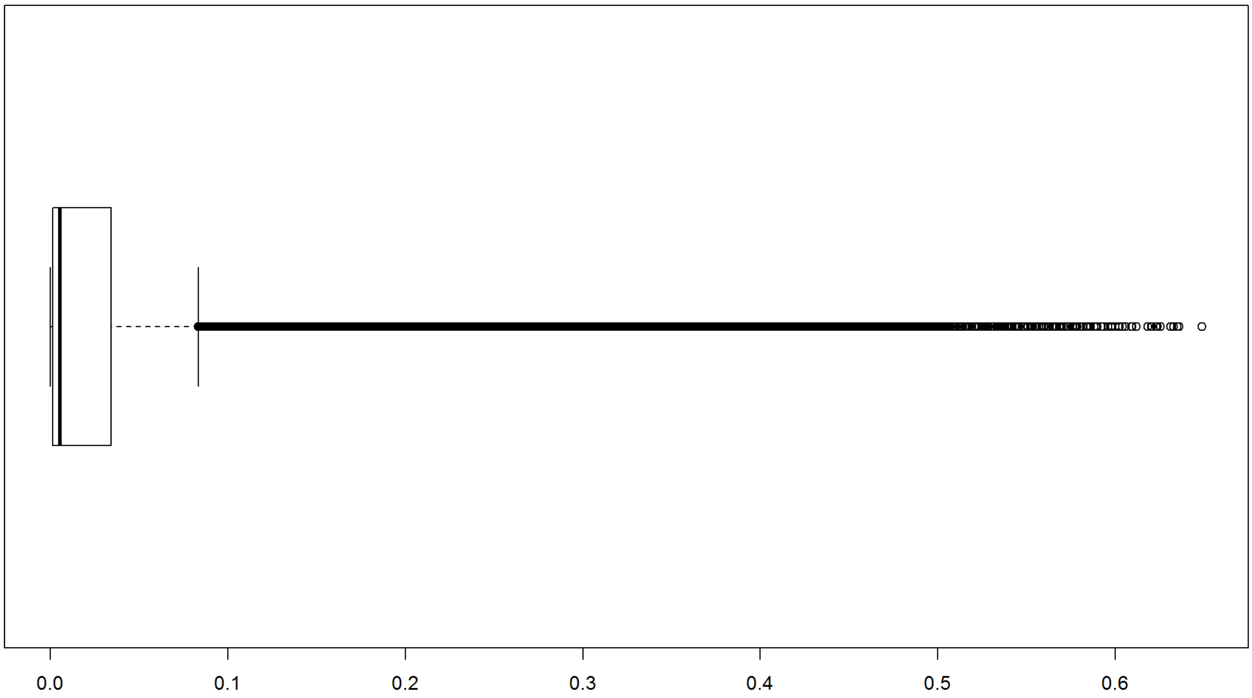
```
## Dim.4  0.88464926  0.1298260 -0.9907032 -1.8073077 -0.7229012 -0.5603507
## Dim.5 -0.04524785 -0.5797503 -0.2338069 -0.8135371 -0.2788809  0.8173275
##             [,7]       [,8]       [,9]
## Dim.1  2.2914168 -2.5022949  1.5985047
## Dim.2 -0.1920530  0.8658863  0.1435979
## Dim.3 -0.8485591  1.6433212  0.1478182
## Dim.4  0.5142000  0.9033222  2.4504885
## Dim.5  0.7971579  1.2553416 -0.1507534
##
## Variances:
## [,,1]
##            Dim.1     Dim.2       Dim.3       Dim.4       Dim.5
## Dim.1  1.4880038 0.3298327 -0.16716791  0.49081804  0.14344023
## Dim.2  0.3298327 1.2814949  0.57331116  0.35400337  0.46001191
## Dim.3 -0.1671679 0.5733112  0.51660921  0.09952362  0.03416554
## Dim.4  0.4908180 0.3540034  0.09952362  0.89982466 -0.03748597
## Dim.5  0.1434402 0.4600119  0.03416554 -0.03748597  0.66178017
## [,,2]
##              Dim.1      Dim.2      Dim.3       Dim.4        Dim.5
## Dim.1  0.439418329 0.55369334 0.16793108  0.46813427 -0.008344946
## Dim.2  0.553693339 0.81268353 0.26695259  0.63406504  0.017131371
## Dim.3  0.167931083 0.26695259 0.11759047  0.20620032  0.005170710
## Dim.4  0.468134273 0.63406504 0.20620032  0.61224067 -0.032608858
## Dim.5 -0.008344946 0.01713137 0.00517071 -0.03260886  0.079758532
## [,,3]
##             Dim.1      Dim.2       Dim.3       Dim.4      Dim.5
## Dim.1  5.14889983 -1.7905766 -0.05424913 -0.65669301  1.9504437
## Dim.2 -1.79057656  3.9964580 -1.02330422  0.46568008 -1.6702216
## Dim.3 -0.05424913 -1.0233042  4.42626727  0.01071176 -0.4474119
## Dim.4 -0.65669301  0.4656801  0.01071176  2.72969893 -2.3507060
## Dim.5  1.95044373 -1.6702216 -0.44741187 -2.35070601  8.9138354
## [,,4]
##            Dim.1     Dim.2      Dim.3     Dim.4      Dim.5
## Dim.1  1.8659284 0.3028319 -0.4165171 0.2332581  0.5094348
## Dim.2  0.3028319 0.8075265  0.3696331 0.6083532  0.2810485
## Dim.3 -0.4165171 0.3696331  0.5516710 0.2561485 -0.1577904
## Dim.4  0.2332581 0.6083532  0.2561485 1.0899806  0.1470555
## Dim.5  0.5094348 0.2810485 -0.1577904 0.1470555  0.5016593
## [,,5]
##               Dim.1       Dim.2         Dim.3       Dim.4       Dim.5
## Dim.1  0.0288076194 -0.01747828 -0.0008022328  0.02824471 -0.04963598
## Dim.2 -0.0174782787  0.05801973  0.0168346545  0.02311488  0.04100444
## Dim.3 -0.0008022328  0.01683465  0.0515012130  0.01302344 -0.06906281
## Dim.4  0.0282447145  0.02311488  0.0130234371  0.12016781 -0.02578051
## Dim.5 -0.0496359763  0.04100444 -0.0690628108 -0.02578051  0.33002261
## [,,6]
##             Dim.1      Dim.2      Dim.3      Dim.4      Dim.5
## Dim.1  0.37197524 -0.2019677 -0.2500266 0.02853637 0.04901762
## Dim.2 -0.20196768  0.3478089  0.3625857 0.14199830 0.15141751
## Dim.3 -0.25002663  0.3625857  0.4245879 0.13988543 0.14810334
## Dim.4  0.02853637  0.1419983  0.1398854 0.13959581 0.11634576
## Dim.5  0.04901762  0.1514175  0.1481033 0.11634576 0.19661086
## [,,7]
##            Dim.1      Dim.2      Dim.3     Dim.4      Dim.5
## Dim.1 0.49034869 0.46598755 0.24243094 0.5022206 0.06229629
## Dim.2 0.46598755 0.59217575 0.27685526 0.5747843 0.08894238
## Dim.3 0.24243094 0.27685526 0.24431227 0.2049949 0.02300664
## Dim.4 0.50222057 0.57478429 0.20499489 0.8146556 0.14663959
## Dim.5 0.06229629 0.08894238 0.02300664 0.1466396 0.08917083
## [,,8]
##             Dim.1      Dim.2      Dim.3     Dim.4       Dim.5
## Dim.1  0.91738451 0.03440719 -0.5931789 0.1601886 -0.35438522
## Dim.2  0.03440719 0.71171383  0.5504263 0.5206445  0.08671913
## Dim.3 -0.59317894 0.55042629  1.0899336 0.5121153  0.55628477
## Dim.4  0.16018855 0.52064451  0.5121153 0.6885001  0.33544012
## Dim.5 -0.35438522 0.08671913  0.5562848 0.3354401  0.69065602
## [,,9]
##              Dim.1       Dim.2        Dim.3       Dim.4        Dim.5
## Dim.1  0.033001333 0.001093541 -0.006271898  0.01381299  0.009502095
## Dim.2  0.001093541 0.071583927  0.037006757  0.03204437  0.036003907
## Dim.3 -0.006271898 0.037006757  0.055215151  0.02447752 -0.006118363
## Dim.4  0.013812991 0.032044372  0.024477518  0.12654093 -0.022250622
## Dim.5  0.009502095 0.036003907 -0.006118363 -0.02225062  0.072595232
```

What is the uncertainty associated with the classification prediction?

```
summary(fit$uncertainty)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.000000 0.001102 0.005195 0.046204 0.033957 0.649018
```

```
boxplot(fit$uncertainty, horizontal = TRUE)
```



There seem to be a high number of outliers, so there may be much skewness in the data.
Interpret clusters

```
gmm.cluster.processed.provider <- cbind(processed.provider,factor(fit$classification))

for (cluster in sort(unique(fit$classification))) {
  temp <- gmm.cluster.processed.provider %>%
    filter(fit$classification == cluster)
  cat("Summarizing cluster: ", cluster)
  cat("\n")
  for (col in names(temp)) {
    if (class(temp[[col]]) != "numerical") {
      print(paste(col, names(which.max(table(temp[[col]]))), sep = ": "))
    }
    else {
      print(paste(col, mean(temp[[col]]), sep = ": "))
    }
  }
  cat("\n")
}
```

```
## Summarizing cluster:  1
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: PHYSICIAN ASSISTANT"
## [1] "State: CA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: No Answer"
```

```
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.2"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(fit$classification): 1"
##
## Summarizing cluster:  2
## [1] "Gender: F"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: NURSE PRACTITIONER"
## [1] "State: TX"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.114285714285714"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(fit$classification): 2"
##
## Summarizing cluster:  3
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Middle"
## [1] "Primary.specialty: CHIROPRACTIC"
## [1] "State: CA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.242857142857143"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(fit$classification): 3"
##
## Summarizing cluster:  4
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: INTERNAL MEDICINE"
## [1] "State: CA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.228571428571429"
## [1] "has.secondary.specialty: Y"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(fit$classification): 4"
##
## Summarizing cluster:  5
## [1] "Gender: F"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: NURSE PRACTITIONER"
## [1] "State: TX"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: No Answer"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.0285714285714286"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(fit$classification): 5"
##
## Summarizing cluster:  6
```

```
## [1] "Gender: F"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: PHYSICAL THERAPY"
## [1] "State: NY"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: No Answer"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.0285714285714286"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: N"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(fit$classification): 6"
##
## Summarizing cluster:  7
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: FAMILY PRACTICE"
## [1] "State: PA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: Y"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.285714285714286"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(fit$classification): 7"
##
## Summarizing cluster:  8
## [1] "Gender: M"
## [1] "Credential: No Answer"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: PHYSICAL THERAPY"
## [1] "State: CA"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: No Answer"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.257142857142857"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: N"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(fit$classification): 8"
##
## Summarizing cluster:  9
## [1] "Gender: M"
## [1] "Credential: MD"
## [1] "Medical.school.name: Top"
## [1] "Primary.specialty: DIAGNOSTIC RADIOLOGY"
## [1] "State: TX"
## [1] "Professional.accepts.Medicare.Assignment: Y"
## [1] "Reported.Quality.Measures: Y"
## [1] "Used.electronic.health.records: No Answer"
## [1] "Committed.to.heart.health.through.the.Million.Hearts..initiative.: No Answer"
## [1] "years.after.grad: 0.285714285714286"
## [1] "has.secondary.specialty: N"
## [1] "has.hospital.affiliation: Y"
## [1] "Number.of.Group.Practice.members_log: 0.532539162864805"
## [1] "factor(fit$classification): 9"
```

Model Comparison

Since we have unlabeled data, the Calinski-Harabaz Index will be used to evaluate the models. The higher the metric, the more dense and well separated the clusters. Source

Network Analysis	K-Means	K-Medoid	Agglomerative Hierarchical Clustering	Gaussian Mixture Model

Network Analysis	K-Means	K-Medoid	Agglomerative Hierarchical Clustering	Gaussian Mixture Model
NA	6741.78156	1.497815210^{5}	NA	5447.8494992

Conclusion

Using the model comparison results above, it is clear that K-Medoid (PAM) has the highest Calinski-Harabaz Index and demonstrates better clustering. However, this data is using 1.330710^{4} observations whereas the other models are using 133061 observations, which skews the validity of this comparison. In the future, it may be helpful to have sampled an even smaller amount from the original population data from CMS.

For this analysis, the Gaussian Mixture Model would be chosen as the best model because the characteristics of this model suits our purpose the most. Indeed, after looking through the PCA results and cluster plots, there is much overlap in clusters which assumes mixed assignment of clusters. Indeed, I suspect that there are too few characteristics/variables in our dataset that could help discern more definitive clusters.

K-Means and K-Medoid could be viable options for a possible model. However, K-Medoid only took a small portion of the data, so this model would be difficult to scale and encompass more provider features and observations.

The network analysis would be interesting to pursue further especially if the full adjacency matrix were used to create the network. Another piece of future work could include adding a binary variable, `included.in.hospital.network`, as a dataset feature.