ProviderScoring

Rohan

```
library(plyr)
```

```
nyopth <- read.csv("nyopth.csv")
agi <- read.csv("agi.csv")
```

```
nyopth.zip = nyopth[,c(2,12)]
func <- function(x){
  substr(x,1,5)
}
nyopth.zip$nppes_provider_zip <- lapply(nyopth.zip$nppes_provider_zip,FUN = func)
nyopth.zip$nppes_provider_zip <- as.numeric(nyopth.zip$nppes_provider_zip)
nyopth.zip = nyopth.zip[!duplicated(nyopth.zip),]
```

```
agi <- agi[,c(1,5)]
```

```
nyopth <- nyopth[,c(2,8,12,16,17,18,20,21,22,23,24,25,26,27,28,29)]

tophcpcs <- names((sort(table(nyopth$hcpcs_code),decreasing = TRUE))[1:10])

spread_scores<-rep(x = 0,times = length(unique(nyopth$npi)))
names(spread_scores) <- unique(nyopth$npi)

bene_scores<-rep(x = 0,times = length(unique(nyopth$npi)))
names(bene_scores) <- unique(nyopth$npi)
```

```
scoreFunction <- function(rowdiff, q, col){
  score = 0

  rowscore = rowdiff[col]
  if (rowscore>=q[1] && rowscore<q[2]){
    score = 0
  }
  else if (rowscore>=q[2] && rowscore<q[3]){
    score = 1
  }
  else if (rowscore>=q[3] && rowscore<q[4]){
    score = 2
  }
  else{
    score = 3
  }
  score
}
```

We loop over all the top hcpcs codes and add to the scores based on the spread of the submitted - payment amount and based on the beneficiaries counts.

```r
for (i in tophcpcs){
  tmp = nyopth[nyopth$hcpcs_code==i,]
  tmp.avgpayment <- ddply(tmp,~npi,summarise,mean=mean(average_medicare_payment_amt))
  tmp.avgsubmitted <- ddply(tmp,~npi,summarise,mean=mean(average_submitted_chrg_amt))
  tmp.spread <- merge(tmp.avgpayment,tmp.avgsubmitted,by.x = "npi",by.y = "npi")
  #TODO:Need to reassess agi index?
  tmp.spread$diff = tmp.spread$mean.y - tmp.spread$mean.x

  t1 = join(x = tmp.spread, y = nyopth.zip, by = "npi")
  colnames(t1)[5] <- "ZIP"
  t2 = join(x = t1, y = agi)

  t2 = transform(t2, diff2 = diff/Indx)
  t2 = t2[complete.cases(t2),]

  #some kind of function that will give out points
  #depending on the quartile that the npi falls under
  q<-quantile(t2$diff2)
  t2$sc <- apply(t2, 1, FUN = scoreFunction,q = q, col = 7)
  for (j in 1:nrow(t2)){
    spread_scores[as.character(t2[j,1])] = spread_scores[as.character(t2[j,1])]+t2[j,8]
  }

  #Dispense points based on number of beneficiaries
  tmp.linecnt <- ddply(tmp, ~npi, summarise, sum=sum(line_srvc_cnt))
  tmp.benecnt <- ddply(tmp, ~npi, summarise, sum=sum(bene_unique_cnt))
  tmp.cntdiff <- merge(tmp.linecnt,tmp.benecnt,by.x = "npi",by.y = "npi")
  tmp.cntdiff$diff = tmp.cntdiff$sum.x / tmp.cntdiff$sum.y

  q2 <- quantile(tmp.cntdiff$diff)
  tmp.cntdiff$sc <- apply(tmp.cntdiff,1,FUN = scoreFunction, q=q2, col = 4)
  for (j in 1:nrow(tmp.cntdiff)){
    bene_scores[as.character(tmp.cntdiff[j,1])] = bene_scores[as.character(tmp.cntdiff[j,1])]+tmp.cntdi
  }
}
```

```
## Joining by: ZIP
## Joining by: ZIP
## Joining by: ZIP
## Joining by: ZIP
## Joining by: ZIP
## Joining by: ZIP
## Joining by: ZIP
## Joining by: ZIP
## Joining by: ZIP
## Joining by: ZIP
```

```r
score_spread <- data.frame(keyName=names(spread_scores), value=spread_scores, row.names=NULL)
score_bene <- data.frame(keyName=names(bene_scores), value=bene_scores, row.names=NULL)

scores <- merge(x=score_spread,y=score_bene, by = "keyName")
colnames(scores) <- c("npi","score_spread","score_bene")
```

```r
summary(scores)
```

```
##          npi          score_spread      score_bene   
##  1003018102:   1   Min.   : 0.000   Min.   : 0.00  
##  1003058652:   1   1st Qu.: 2.000   1st Qu.: 6.00  
##  1003812454:   1   Median : 8.000   Median :10.00  
##  1003813247:   1   Mean   : 9.234   Mean   :10.69  
##  1003814682:   1   3rd Qu.:15.000   3rd Qu.:15.00  
##  1003815432:   1   Max.   :30.000   Max.   :26.00  
##  (Other)   :1608                                   
```