

US Hospital Comapre-Finding Efficiency Hospital

Chunhui Zhu

October 4, 2017

Assignment

The goal of this assignment is to give you practice in preparing different datasets for downstream analysis work.

Your task is to:

(1)Choose any three of the “wide” datasets identified in the Week 6 Discussion items. (You may use your own dataset; please don’t use my Sample Post dataset, since that was used in your Week 6 assignment!)

For each of the three chosen datasets: ???Create a .CSV file (or optionally, a MySQL database!) that includes all of the information included in the dataset. You’re encouraged to use a “wide” structure similar to how the information appears in the discussion item, so that you can practice tidying and transformations as described below. ??? Read the information from your .CSV file into R, and use tidyr and dplyr as needed to tidy and transform your data. [Most of your grade will be based on this step!] ??? ####(2)Perform the analysis requested in the discussion item. ??? ####our code should be in an R Markdown file, posted to rpubs.com, and should include narrative descriptions of your data cleanup work, analysis,and conclusions.

+++++

NY state hospital Timely and Effective Care Evalution

Project Period: 10/1/2015-9/30/2016

In this porject, we are going to visualize how timely and effective for states local hospitls’ performance based emergency care.The eveluation will depent on the waiting time(Score) and percentage (Score) in comparision. The goal is to find the best efficient hospital in each by various category, and in overall category.

R “tidyr” “dplyr”

```
library("tidyr")
library("dplyr")
```

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

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

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

```
library("ggplot2")
library("ggthemes")
```

```
## Warning: package 'ggthemes' was built under R version 3.4.2
```

US Timely and Effective Care data set Resource

<https://data.medicare.gov/data/hospital-compare#>

Date Set Description:

“The measures of timely and effective care (also known as “process of care” measures) show the percentage of hospital patients who got treatments known to get the best results for certain common, serious medical conditions or surgical procedures; how quickly hospitals treat patients who come to the hospital with certain medical emergencies; and how well hospitals provide preventive services. These measures only apply to patients for whom the recommended treatment would be appropriate. The measures of timely and effective care apply to adults and children treated at hospitals paid under the Inpatient Prospective Payment System (IPPS) or the Outpatient Prospective Payment System (OPPS), as well as those that voluntarily report data on measures for whom the recommended treatments would be appropriate including: Medicare patients, Medicare managed care patients, and non-Medicare patients. Timely and effective care measures include cataract care follow-up, colonoscopy follow-up, heart attack care, emergency department care, preventive care, stroke care, blood clot prevention and treatment, and pregnancy and delivery care measures. ”

Three data sets:

1.table ndf-Timely and Effective Care - National

2.table sdf-Timely and Effective Care - State

3.table hdf-Timely and Effective Care - Hospital

There are some common variables in these three tables, “Measure Name”, “Measure ID”, “Score”, “Condition”, “Measure Start Date” and “Measure End Date”. Among of them, “Measure ID” is briefs/symbol of “Measure Name”, giving the description of observation. “Score” can be waiting time or percentage. “Condition” is department of patient visit, like Emergency Department, Heart Attack or Chest Pain, Preventive Care, and etc. “Measure Start Date” and “Measure End Date” provide us the period from Oct.1 2015 to Sep.30 2016.

```
ndf<-read.csv("Timely and Effective Care _National.csv",stringsAsFactors = FALSE)
glimpse(ndf)
```

```
## Observations: 41
## Variables: 8
## $ Measure.Name      <chr> "Average (median) time patients spent in th...
## $ Measure.ID        <chr> "ED_1b", "ED_1b_HIGH_MIN", "ED_1b_LOW_MIN",...
## $ Condition         <chr> "Emergency Department", "Emergency Departme...
## $ Category          <chr> "ED1", "ED1 - Median time from emergency de...
## $ Score              <int> 279, 295, 213, 258, 275, 333, 100, 118, 58,...
## $ Footnote          <chr> "20 - State and national averages do not in...
## $ Measure.Start.Date <chr> "10/01/2015", "10/01/2015", "10/01/2015", "...
## $ Measure.End.Date  <chr> "09/30/2016", "09/30/2016", "09/30/2016", "...
```

```
sdf<-read.csv("Timely and Effective Care_State.csv",stringsAsFactors = FALSE)
glimpse(sdf)

## Observations: 2,296
## Variables: 8
## $ State <chr> "AK", "AL", "AR", "AS", "AZ", "CA", "CO", "...
## $ Condition <chr> "Heart Attack or Chest Pain", "Heart Attack...
## $ Measure.Name <chr> "Average (median) number of minutes before ...
## $ Measure.ID <chr> "OP_3b", "OP_3b", "OP_3b", "OP_3b", "OP_3b"...
## $ Score <chr> "262", "66", "58", "Not Available", "60", "...
## $ Footnote <chr> "20 - State and national averages do not in...
## $ Measure.Start.Date <chr> "10/01/2015", "10/01/2015", "10/01/2015", "...
## $ Measure.End.Date <chr> "09/30/2016", "09/30/2016", "09/30/2016", "...

hdf<-read.csv("Timely and Effective Care _Hospital.csv",stringsAsFactors = FALSE)
glimpse(hdf)

## Observations: 105,864
## Variables: 16
## $ Provider.ID <int> 10001, 10001, 10001, 10001, 10001, 10001, 1...
## $ Hospital.Name <chr> "SOUTHEAST ALABAMA MEDICAL CENTER", "SOUTHE...
## $ Address <chr> "1108 ROSS CLARK CIRCLE", "1108 ROSS CLARK ...
## $ City <chr> "DOTHAN", "DOTHAN", "DOTHAN", "DOTHAN", "DO...
## $ State <chr> "AL", "AL", "AL", "AL", "AL", "AL", "AL", "...
## $ ZIP.Code <int> 36301, 36301, 36301, 36301, 36301, 36301, 3...
## $ County.Name <chr> "HOUSTON", "HOUSTON", "HOUSTON", "HOUSTON",...
## $ Phone.Number <dbl> 3347938701, 3347938701, 3347938701, 3347938...
## $ Condition <chr> "Emergency Department", "Emergency Departme...
## $ Measure.ID <chr> "ED_1b", "ED_2b", "EDV", "IMM_2", "IMM_3_OP...
## $ Measure.Name <chr> "ED1", "ED2", "Emergency department volume"...
## $ Score <chr> "279", "89", "High (40,000 - 59,999 patient...
## $ Sample <chr> "559", "559", "", "554", "3791", "Not Avail...
## $ Footnote <chr> "2 - Data submitted were based on a sample ...
## $ Measure.Start.Date <chr> "10/01/2015", "10/01/2015", "01/01/2015", "...
## $ Measure.End.Date <chr> "09/30/2016", "09/30/2016", "12/31/2015", "...
```

Step1:Join tables and rename columns

My method is to use left_join function to connect these three tables. In table hdf,I observed 'measure ID' in hdf and sdf tables has some observations that are not in hdf, like min, max, high_min, low_min, high_max and ect.For example, "ED_1b_MEDIUM_MIN" gives "ED_1b" lower bound of medium at the same measure.After joined three table, these observations will have a lot of NA value under hospital's variables.

```
sumdf<-left_join(left_join(hdf, sdf, by=c("Measure.Name","Measure.ID","Condition","Footnote","Measure.S...
dim(sumdf)

## [1] 100326      21
glimpse(sumdf)

## Observations: 100,326
## Variables: 21
## $ Measure.Name.x <chr> "Average (median) time patients spent in th...
## $ Measure.ID <chr> "ED_1b", "ED_1b", "ED_1b", "ED_1b", "ED_1b"...
```

```
## $ Condition      <chr> "Emergency Department", "Emergency Departme...
## $ Category       <chr> "ED1", "ED1", "ED1", "ED1", "ED1", "ED1", "...
## $ Score.x        <int> 279, 279, 279, 279, 279, 279, 279, 279, 279...
## $ Footnote.x     <chr> "20 - State and national averages do not in...
## $ Measure.Start.Date <chr> "10/01/2015", "10/01/2015", "10/01/2015", "...
## $ Measure.End.Date <chr> "09/30/2016", "09/30/2016", "09/30/2016", "...
## $ State          <chr> "AK", "AK", "AK", "AK", "AK", "AK", "AK", "...
## $ Score.y        <chr> "303", "303", "303", "303", "303", "303", "...
## $ Provider.ID    <int> 20001, 20006, 20008, 20012, 20017, 20018, 2...
## $ Hospital.Name  <chr> "PROVIDENCE ALASKA MEDICAL CENTER", "MAT-SU...
## $ Address        <chr> "3200 PROVIDENCE DRIVE", "2500 SOUTH WOODWO...
## $ City           <chr> "ANCHORAGE", "PALMER", "JUNEAU", "FAIRBANKS...
## $ ZIP.Code       <int> 99508, 99645, 99801, 99701, 99508, 99559, 9...
## $ County.Name    <chr> "ANCHORAGE", "MATANUSKA SUSITNA", "JUNEAU",...
## $ Phone.Number   <dbl> 9075622211, 9078616000, 9077968900, 9074528...
## $ Measure.Name.y <chr> "ED1", "ED1", "ED1", "ED1", "ED1", "ED1", "...
## $ Score          <chr> "404", "261", "310", "294", "266", "309", "...
## $ Sample         <chr> "530", "685", "306", "404", "658", "291", "...
## $ Footnote.y     <chr> "2 - Data submitted were based on a sample ...
```

Rename and clarify variable names in sumdf data set.

```
colnames(sumdf)[colnames(sumdf)=="Measure.Name.x"] <- "Measure.Name.National"
colnames(sumdf)[colnames(sumdf)=="Footnote.x"] <- "Footnote.National"
colnames(sumdf)[colnames(sumdf)=="Score.x"] <- "Score.National"
colnames(sumdf)[colnames(sumdf)=="Score.y"] <- "Score.State"
colnames(sumdf)[colnames(sumdf)=="Score"] <- "Score.Hospital"
colnames(sumdf)[colnames(sumdf)=="Measure.Name.y"] <- "Measure.Name.Hospital"
colnames(sumdf)[colnames(sumdf)=="Sample"] <- "Sample.Hospital"
colnames(sumdf)[colnames(sumdf)=="Footnote.y"] <- "Footnote.Hospital"
head(sumdf,3)
```

```
##
## 1 Average (median) time patients spent in the emergency department, before they were admitted to the
## 2 Average (median) time patients spent in the emergency department, before they were admitted to the
## 3 Average (median) time patients spent in the emergency department, before they were admitted to the
## Measure.ID Condition Category Score.National
## 1 ED_1b Emergency Department ED1 279
## 2 ED_1b Emergency Department ED1 279
## 3 ED_1b Emergency Department ED1 279
## Footnote.National
## 1 20 - State and national averages do not include VHA hospital data.
## 2 20 - State and national averages do not include VHA hospital data.
## 3 20 - State and national averages do not include VHA hospital data.
## Measure.Start.Date Measure.End.Date State Score.State Provider.ID
## 1 10/01/2015 09/30/2016 AK 303 20001
## 2 10/01/2015 09/30/2016 AK 303 20006
## 3 10/01/2015 09/30/2016 AK 303 20008
## Hospital.Name Address City
## 1 PROVIDENCE ALASKA MEDICAL CENTER 3200 PROVIDENCE DRIVE ANCHORAGE
## 2 MAT-SU REGIONAL MEDICAL CENTER 2500 SOUTH WOODWORTH LOOP PALMER
## 3 BARTLETT REGIONAL HOSPITAL 3260 HOSPITAL DR JUNEAU
## ZIP.Code County.Name Phone.Number Measure.Name.Hospital
## 1 99508 ANCHORAGE 9075622211 ED1
## 2 99645 MATANUSKA SUSITNA 9078616000 ED1
```

```
## 3      99801          JUNEAU  9077968900          ED1
##   Score.Hospital Sample.Hospital
## 1          404          530
## 2          261          685
## 3          310          306
##                                     Footnote.Hospital
## 1 2 - Data submitted were based on a sample of cases/patients.
## 2 2 - Data submitted were based on a sample of cases/patients.
## 3 2 - Data submitted were based on a sample of cases/patients.

creat a hospital_US.csv file for t data set at "C:/Users/Ivy/Desktop/607/W6"

setwd("C:/Users/Ivy/Desktop/607/W6")
write.csv(sumdf,"hospital_US.csv")
```

Step2: Chuncate NA data set and Change data type

I am going to remove some observations. For example, "ED_1b_MEDIUM_MIN" gives "ED_1b" lower bound of medium at the same mesurement. I will remove such type of observations. Also, I will remove the "Not Available" in Score.Hospital and Sample.Hospital columns.

```
sumdf1<-filter(sumdf, nchar(sumdf$Measure.ID)<7)
sumdf1<-sumdf1[- grep("Not Available", sumdf1$Score.Hospital),]
glimpse(sumdf1)
```

```
## Observations: 45,274
## Variables: 21
## $ Measure.Name.National <chr> "Average (median) time patients spent in...
## $ Measure.ID            <chr> "ED_1b", "ED_1b", "ED_1b", "ED_1b", "ED_...
## $ Condition             <chr> "Emergency Department", "Emergency Depar...
## $ Category              <chr> "ED1", "ED1", "ED1", "ED1", "ED1", "ED1"...
## $ Score.National        <int> 279, 279, 279, 279, 279, 279, 279, 279, ...
## $ Footnote.National     <chr> "20 - State and national averages do not...
## $ Measure.Start.Date    <chr> "10/01/2015", "10/01/2015", "10/01/2015"...
## $ Measure.End.Date      <chr> "09/30/2016", "09/30/2016", "09/30/2016"...
## $ State                 <chr> "AK", "AK", "AK", "AK", "AK", "AK", "AK"...
## $ Score.State           <chr> "303", "303", "303", "303", "303", "303"...
## $ Provider.ID           <int> 20001, 20006, 20008, 20012, 20017, 20018...
## $ Hospital.Name         <chr> "PROVIDENCE ALASKA MEDICAL CENTER", "MAT...
## $ Address               <chr> "3200 PROVIDENCE DRIVE", "2500 SOUTH WOO...
## $ City                  <chr> "ANCHORAGE", "PALMER", "JUNEAU", "FAIRBA...
## $ ZIP.Code              <int> 99508, 99645, 99801, 99701, 99508, 99559...
## $ County.Name           <chr> "ANCHORAGE", "MATANUSKA SUSITNA", "JUNEA...
## $ Phone.Number          <dbl> 9075622211, 9078616000, 9077968900, 9074...
## $ Measure.Name.Hospital <chr> "ED1", "ED1", "ED1", "ED1", "ED1", "ED1"...
## $ Score.Hospital        <chr> "404", "261", "310", "294", "266", "309"...
## $ Sample.Hospital       <chr> "530", "685", "306", "404", "658", "291"...
## $ Footnote.Hospital     <chr> "2 - Data submitted were based on a samp..."
```

I convert Score and sample's char data type to numeric data type. Also I observed "Score" could be median or percentages. I divide different type of Score into two tables median score and percent score. I filter condition which is emergency department in two tables Emg_Median and Emg_percent.

```
sumdf1$Score.State<-as.numeric(sumdf1$Score.State)
sumdf1$Score.Hospital<-as.numeric(sumdf1$Score.Hospital)
sumdf1$Sample.Hospital<-as.numeric(sumdf1$Sample.Hospital)
```

```

v<-grep("Median|median", sumdf1$Measure.Name.National,value = TRUE)
median_t<-sumdf1[sumdf1$Measure.Name.National %in% v,]
Emg_median<-filter(median_t, median_t$Condition=="Emergency Department")
#View(Emg_median)

percent_t<-subset(sumdf1,! (sumdf1$Measure.Name.National %in% v))
percent_t$Score.National<-percent_t$Score.National/100
percent_t$Score.State<-percent_t$Score.State/100
percent_t$Score.Hospital<-percent_t$Score.Hospital/100
Emg_percent<-filter(percent_t, percent_t$Condition=="Emergency Department")
#View(Emg_percent)

```

Step3: Seperate Score: interger and percentage

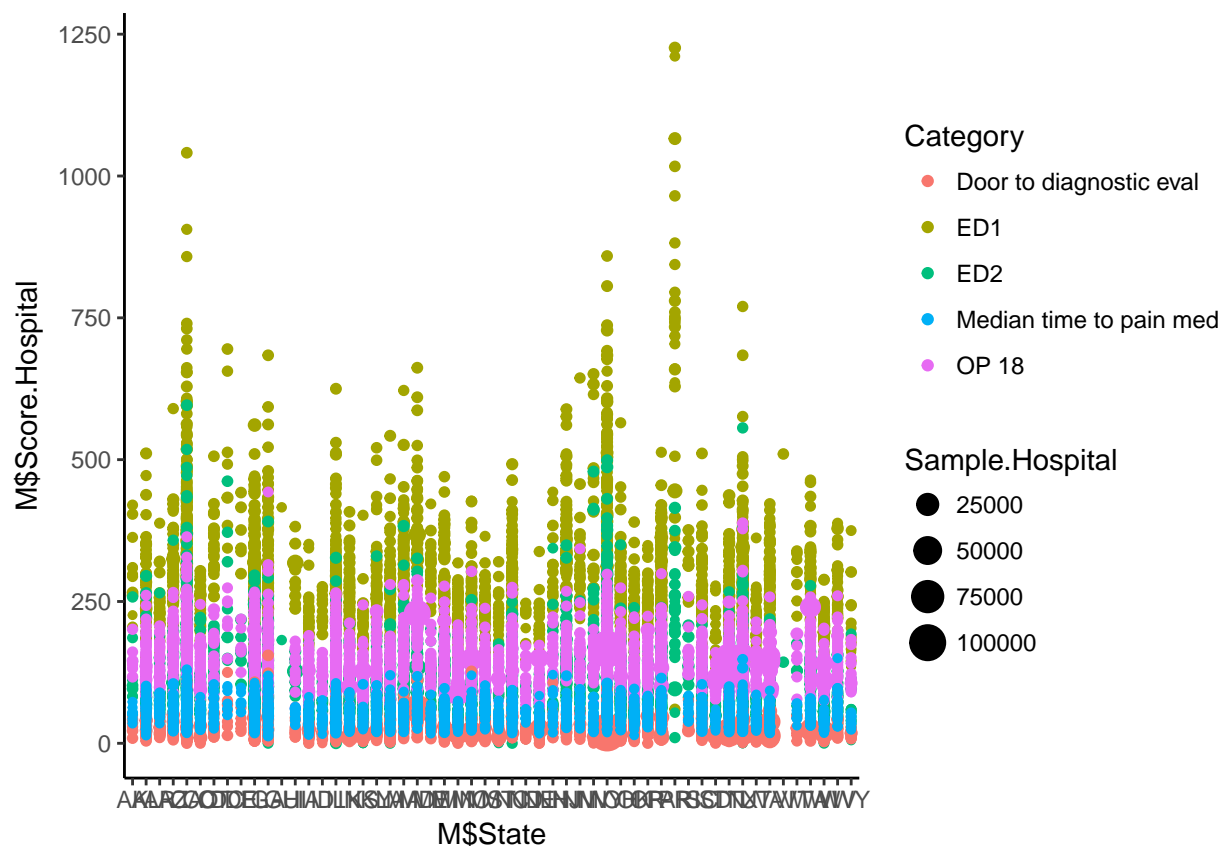
I creat two subset from two tables and named M (Median Score) and P(Percentage Score). Each contains 6 variables "Hospital.Name", "Category", "State,Score.State", "Score.Hospital", "Sample.Hospital".

The following will look for the spread of local hospitals in each state by different category.

```

M<-select(Emg_median,Hospital.Name,Category,State,Score.State,Score.Hospital,Sample.Hospital)
#summary(M)
ggplot(M,aes(x=M$State,y=M$Score.Hospital,color=Category, size=Sample.Hospital))+geom_point()+theme_cla

```



Discription for Category:

Door to diagnostic eval: Average (median) time patients spent in the emergency department before they were seen by a healthcare professional A lower number of minutes is better

ED1:Average (median) time patients spent in the emergency department, before they were admitted to the hospital as an inpatient A lower number of minutes is better

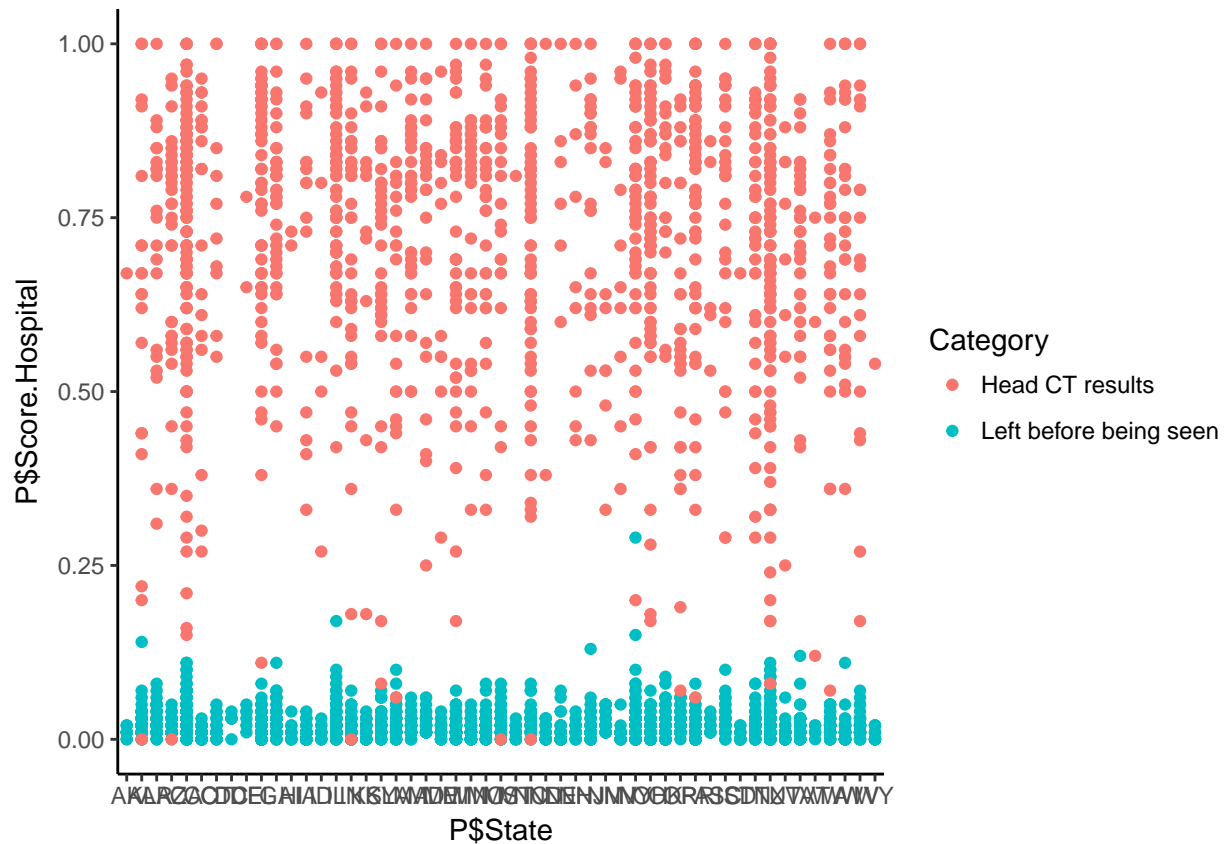
ED2:Average (median) time patients spent in the emergency department, after the doctor decided to admit them as an inpatient before leaving the emergency department for their inpatient room A lower number of minutes is better

Median time to pain med: Average (median) time patients who came to the emergency department with broken bones had to wait before getting pain medication A lower number of minutes is better

Average (median) time patients spent in the emergency department before leaving from the visit A lower number of minutes is better

All measures in category are within a range of the score. In table M, all measures are the lower score are the better.

```
P<-select(Emg_percent,Hospital.Name,Category,State,Score.State,Score.Hospital,Sample.Hospital)
#summary(P)
ggplot(P,aes(x=P$State,y=P$Score.Hospital,color=Category))+geom_point()+theme_classic()
```



####Discription for Category:

Head CT results: Percentage of patients who came to the emergency department with stroke symptoms who received brain scan results within 45 minutes of arrival Higher percentages are better

Left before being seen: Percentage of patients who left the emergency department before being seen Lower percentages are better

All measures in category are within a range of the score. In table M, all measures are the min score are the best.

Step5: Evaluate correlation of Score.Hospital and Sample.Hospital

The following I will use cor function to find the correlation.

```
cor(M$Score.Hospital,M$Sample.Hospital)

## [1] 0.05579665

Mt<-split(M, with(M, interaction(Category)), drop = F)
cor(Mt$`Door to diagnostic eval`$Score.Hospital, Mt$`Door to diagnostic eval`$Sample.Hospital)

## [1] -0.003217092

cor(Mt$ED1$Score.Hospital, Mt$ED1$Sample.Hospital)

## [1] 0.3599023

cor(Mt$ED2$Score.Hospital, Mt$ED2$Sample.Hospital)

## [1] 0.3731935

cor(Mt$`Median time to pain med`$Score.Hospital, Mt$`Median time to pain med`$Sample.Hospital)

## [1] 0.090303

cor(Mt$`OP 18`$Score.Hospital, Mt$`OP 18`$Sample.Hospital)

## [1] 0.02319498

cor(P$Score.Hospital,P$Sample.Hospital)

## [1] -0.4660535

Pt<-split(P, with(P, interaction(Category)), drop = F)
cor(Pt$`Head CT results`$Score.Hospital, Pt$`Head CT results` $Sample.Hospital)

## [1] 0.1396918

cor(Pt$`Left before being seen` $Score.Hospital, Pt$`Left before being seen` $Sample.Hospital)

## [1] 0.2197447
```

From the correlation coefficient, Score.Hospital and Sample.Hospital have no correlation or very weak correlation.

Step6:Scale sample size in Score, and find the most efficient hospital in each state by category

To evaluate the performance for each hospital, I will scale the sample size in the score.hospital, which mean the one handling larger size, the one will have better score.

For Median score talbe, I calculate the control value which is ration of each sample.hospital and the mean size of each state. Since in this table the small number is the better. so I use each Score.Hospital divide the control value.

```
M1<- M%>% group_by(Category,State)%>%mutate(mean(Sample.Hospital))%>%mutate(Score.Hospital/(Sample.Hosp

colnames(M1)[colnames(M1)=="Score.Hospital/(Sample.Hospital/mean(Sample.Hospital))"] <- "Socre.H.StateM
```



```
M2<-M1%>%group_by(Category,State) %>% filter(Socre.H.StateM==min(Socre.H.StateM))
BestM<-data.frame(M2$Hospital.Name,M2$Category,M2$State)
colnames(BestM)<-c("Hospital.Name","Category","State")
```

For Percent score talbe, I evaluate “Head CT results”.I calculate the control value which is ration of each sample.hospital and the mean size of each state. Since in this table the higher number is the better. so I use each Score.Hospital multiply the control value.

```
P1<-P%>% filter(Category=="Head CT results")%>% group_by(State)%>%mutate(mean(Sample.Hospital))%>%mutate(
colnames(P1)[colnames(P1)=="Score.Hospital * (Sample.Hospital/mean(Sample.Hospital))"] <- "Socre.H.StateM")
```

```
P11<- P1%>%group_by(State) %>% filter(Socre.H.StateM==max(Socre.H.StateM))
BestP1<-data.frame(P11$Hospital.Name,P11$Category,P11$State)
colnames(BestP1)<-c("Hospital.Name","Category","State")
```

```
BestP1%>%group_by(Hospital.Name,Category)%>%filter(n()>1) #Check if there duplicate value after group_by
```

```
## # A tibble: 0 x 3
## # Groups:   Hospital.Name, Category [0]
## # ... with 3 variables: Hospital.Name <fctr>, Category <fctr>,
## #   State <fctr>
```

For Percent score talbe, I evaluate “Left before being seen”.I calculate the control value which is ration of each sample.hospital and the mean size of each state. Since in this table the lower number is the better. so I use each Score.Hospital divide the control value.

```
P2<-P%>% filter(Category=="Left before being seen")%>% group_by(State)%>%mutate(mean(Sample.Hospital))%>%
colnames(P2)[colnames(P2)=="Score.Hospital/(Sample.Hospital/mean(Sample.Hospital))"] <- "Socre.H.StateM")
```

```
P22<- P2%>%group_by(State) %>% filter(Socre.H.StateM==max(Socre.H.StateM))
BestP2<-data.frame(P22$Hospital.Name,P22$Category,P22$State)
colnames(BestP2)<-c("Hospital.Name","Category","State")
```

```
BestP2%>%group_by(Hospital.Name,Category)%>%filter(n()>1) #Check if there duplicate value after group_by
```

```
## # A tibble: 0 x 3
## # Groups:   Hospital.Name, Category [0]
## # ... with 3 variables: Hospital.Name <fctr>, Category <fctr>,
## #   State <fctr>
```

The Efficient_CS table includes the most efficient performtent hospital by category in each state.

```
Efficient_CS<-bind_rows(BestM,BestP1,BestP2)
```

```
## Warning in bind_rows(x, .id): Unequal factor levels: coercing to character
## Warning in bind_rows(x, .id): binding character and factor vector,
## coercing into character vector
```

```
## Warning in bind_rows(x, .id): binding character and factor vector,
## coercing into character vector
```

```
## Warning in bind_rows(x, .id): Unequal factor levels: coercing to character
```

```
## Warning in bind_rows(x, .id): binding character and factor vector,
## coercing into character vector
```

```
## Warning in bind_rows(x, .id): binding character and factor vector,
## coercing into character vector

## Warning in bind_rows(x, .id): Unequal factor levels: coercing to character

## Warning in bind_rows(x, .id): binding character and factor vector,
## coercing into character vector

## Warning in bind_rows(x, .id): binding character and factor vector,
## coercing into character vector

## Warning in bind_rows(x, .id): binding character and factor vector,
## coercing into character vector

## Warning in bind_rows(x, .id): binding character and factor vector,
## coercing into character vector
```

```
Efficient_CS%>%group_by(Hospital.Name,Category)%>%filter(n()>1)
```

```
## # A tibble: 0 x 3
## # Groups:   Hospital.Name, Category [0]
## # ... with 3 variables: Hospital.Name <chr>, Category <chr>, State <chr>
#Check if there duplicate value after group_by columns, 0 means no duplicate.
```

```
head(Efficient_CS)
```

```
##           Hospital.Name Category State
## 1  MAT-SU REGIONAL MEDICAL CENTER    ED1    AK
## 2  MARSHALL MEDICAL CENTER SOUTH    ED1    AL
## 3  WHITE COUNTY MEDICAL CENTER    ED1    AR
## 4  CANYON VISTA MEDICAL CENTER    ED1    AZ
## 5 MENDOCINO COAST DISTRICT HOSPITAL    ED1    CA
## 6          COMMUNITY HOSPITAL    ED1    CO
```

Step7:Scale sample size in Score, and find the overall most efficient hospital in each state

Since “Head CT results” in P1 category have negative relation with the other measures, I use its score multiply -1 before I sum up all other scores.

```
P1$Socre.H.StateM<-P1$Socre.H.StateM*(-1)
```

```
BestAllM<-bind_rows(M1,P1,P2)
```

```
BestAllM<-M1%>%group_by(State,Hospital.Name)%>%summarise(Socre.H.StateM = sum(Socre.H.StateM))%>%filter(
head(BestAllM))
```

```
## # A tibble: 6 x 3
## # Groups:   State [6]
##   State           Hospital.Name Socre.H.StateM
##   <chr>           <chr>           <dbl>
## 1  AK              MT EDGE CUMBE HOSPITAL    296.1687
## 2  AL              MARSHALL MEDICAL CENTER SOUTH    187.6526
## 3  AR              RIVER VALLEY MEDICAL CENTER    99.5877
## 4  AZ              CANYON VISTA MEDICAL CENTER    311.3138
```

## 5	CA KAISER FOUNDATION HOSPITAL - SOUTH SAN FRANCISCO	243.0406
## 6	CO MEMORIAL HOSPITAL, THE	210.1025