# Notes for HCUP NEDS Analysis

Charles DiMaggio
Bellevue - NYU Trauma Service

February 14, 2017

## Contents

**1 Introduction** **2**

**2 Required R Libraries** **2**

**3 Analyses** **4**
3.1 Descriptive Statistics . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
3.1.1 Age and Gender . . . . . . . . . . . . . . . . . . . . . . . . . 6
3.1.2 Deaths . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
3.2 Injury Severity and Comorbidities . . . . . . . . . . . . . . . . . . . . . 9
3.3 Hospital Type . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13
3.4 Injury Categories and Causes . . . . . . . . . . . . . . . . . . . . . . . 16
3.5 Costs . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20

**4 File Creation** **22**
4.1 Read csv's into R . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
4.2 Select Out Injuries . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26
4.3 Add Barell Matrix Diagnostic Codes . . . . . . . . . . . . . . . . . . . . 30
4.4 Calculate and Add ICD-Derived Injury Severity Scores (ICISS) . . . . . . . . 38
4.5 Calculate Charlson Index . . . . . . . . . . . . . . . . . . . . . . . . . . 47
4.6 Add Trauma Center Variable . . . . . . . . . . . . . . . . . . . . . . . . 48
4.7 Calculate Cost Variable . . . . . . . . . . . . . . . . . . . . . . . . . . . 49
4.8 Write MonetDB File . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 50

**5 Appendix: Survey Procedures Run on Full R Data Set** **52**

# 1 Introduction

These code notes are intended to help interested researchers reproduce published analyses of HCUP NEDS data. Most folks will find themselves to these notes via a link in a journal article. [1] They should be adequate to get folks well on their way. The full set of code runs to about 5,000 lines and translates to a 120-page pdf document. It includes updated analyses in response to peer reviewers, alternate approaches, failed runs, etc... If you need that level of detail, please contact me.

# 2 Required R Libraries

Most of the analyses use the "survey" package. The "multicore" package allows the "multicore=T" option in some survey procedures and must be downloaded from the R archive. "MonetDB" and "plyr" allow out-of-memory storage and access of the large NEDS dataset.

```
library(knitr)
library(survey)
options(survey.lonely.psu = "adjust")
library(ggplot2)
# take advantage of multicore=T option in survey
# install.packages('/Users/charlie/Downloads/multicore_0.2.tar.gz',
# repos = NULL, type='source') after downloading from
# https://cran.r-project.org/src/contrib/Archive/multicore/
library(multicore)
options(survey.multicore = TRUE)  # multiple processors by default

library(ggplot2)
library(data.table)
library(icd9)
library(xtable)
library(cowplot)  # for plot_grid
library(rms)  # adjusted regression

library(DBI)
library(MonetDB.R)
library(MonetDBLite)
library(plyr)
library(dplyr)

# install.packages('sqlsurvey',
# repos='http://R-Forge.R-project.org') after downloading
# from
# https://cran.r-project.org/src/contrib/Archive/multicore/
```

[1] If you stumbled onto this through some kind of search, enjoy. But if you don't know what HCUP or NEDS are, this probably isn't the place for you.

```r
library(sqlsurvey)

inj <- readRDS("~/neds_06_12_inj.rds")
```

Establish a connection to the out-of-memory Monet database containing the NEDS injury file.[2] Check the file and fields.

```r
mdb <- dbConnect(MonetDBLite(), "~/monetInjury")
dbListTables(mdb)
dbListFields(mdb, "nedsinj0612")
dbGetQuery(mdb, "SELECT COUNT(*) FROM nedsinj0612") # 40073358 observations (12.31 GB)
```

Source and create a dplyr table from MonetDB file.

```r
mdb_src <- src_monetdb(embedded = "~/monetInjury")
nedsinj <- tbl(mdb_src, "nedsinj0612")
head(nedsinj)
nrow(nedsinj)  # 40073358
str(nedsinj)  # returns structure of the connection object
glimpse(nedsinj)  # dplyr fx returns neater structure of the table or df
mean(nedsinj$AGE)  # base fails
summarise(nedsinj, mean_age = mean(age))  # dplyr works
count(nedsinj, hosp_region)
count(nedsinj, severe)
```

Create a sqlsurvey object from the MonetDB injury table.[3] for details. Note the speed with which the survey object is created and statistics can be run on this very large file.

```r
fax <- c("aweekend", "died_visit", "female", "intent_self_harm",
    "injury", "injury_cut", "injury_drown", "injury_fall", "injury_fire",
    "injury_firearm", "injury_machinery", "injury_mvt", "injury_nature",
    "injury_poison", "injury_severity", "injury_struck", "injury_suffocation",
    "intent_assault", "intent_unintentiona", "multinjury", "intent_unintention",
    "hosp_trauma", "hosp_ur_teach")

system.time(monInj <- sqlsurvey(id = "key_ed", strata = "neds_stratum",
   weights = "discwt", key = "id", check.factors = fax, database = "/Volumes/Promise Pegasus/DATA/HCUI
    driver = MonetDBLite(), table.name = "nedsinj0612"))

# user system elapsed 10.412 9.854 64.503
```

[2]See section 3 for file creation code.

[3]See http://rpackages.ianhowson.com/rforge/sqlsurvey/ and http://rpackages.ianhowson.com/rforge/sqlsurvey/man/sqlsurvey.html

# 3 Analyses

## 3.1 Descriptive Statistics

There were 181,194,431 (s.e. = 4,234) emergency department visits for traumatic injury in the United States between 2006 and 2012. Table 1 There was an approximately 9% decrease in the rate of emergency department visits for traumatic injury per 100,000 population over the study period, from 8,923 (95% CI 8,916, 8,930) in 2006 to 8,115 (95% CI 8,109, 8,121) in 2012. Figure 1

| Year | Count (s.e.) |
|---|---|
| 2006 | 26,624,017 (10,381) |
| 2007 | 26,519,522 (10,448) |
| 2008 | 26,084,855 ( 9,888) |
| 2009 | 25,376,355 (10,296) |
| 2010 | 25,771,892 ( 9,933) |
| 2011 | 25,347,338 ( 8,907) |
| 2012 | 25,470,453 ( 9,400) |
| Total | 181,194,431 (4,234) |

Table 1: Emergency Department Visits for Traumatic Injury, US Hospitals, 2006-2012.

```
yrCount <- svytotal(~count, monInj, byvar = ~yr, na.rm = T, se = TRUE,
    multicore = T)
print(yrCount)

# plot total rate over time pops from MS2014-02 Population
# Denominator Data Tables - Appendix A 08152014.xlsx
natPop <- c(313873685, 311582564, 309326295, 306771529, 304093966,
    301231207, 298379912)
natPop <- natPop[7:1]  # arrange in increasing years
count <- c(26624017, 26519522, 26084855, 25376355, 25771892,
    25347338, 25470453)
totCount_sd <- c(10381.19, 10448.574, 9888.642, 10296.107, 9933.642,
    8907.878, 9400.411)
lowerCount <- count - (1.96 * totCount_sd)
upperCount <- count + (1.96 * totCount_sd)
natRate <- data.frame(Rate = count/natPop * 1e+05, lowerRate = lowerCount/natPop *
    1e+05, upperRate = upperCount/natPop * 1e+05, Year = 2006:2012)

(natRate[1, 1] - natRate[7, 1])/natRate[1, 1] * 100

# NB: upper and lower ci's not appreciable on plot so
# omitting
p1 <- ggplot(data = natRate, aes(x = Year, y = Rate))
p2 <- p1 + geom_line(aes(group = 1)) + ylim(0, 10000) + xlim(2006,
```

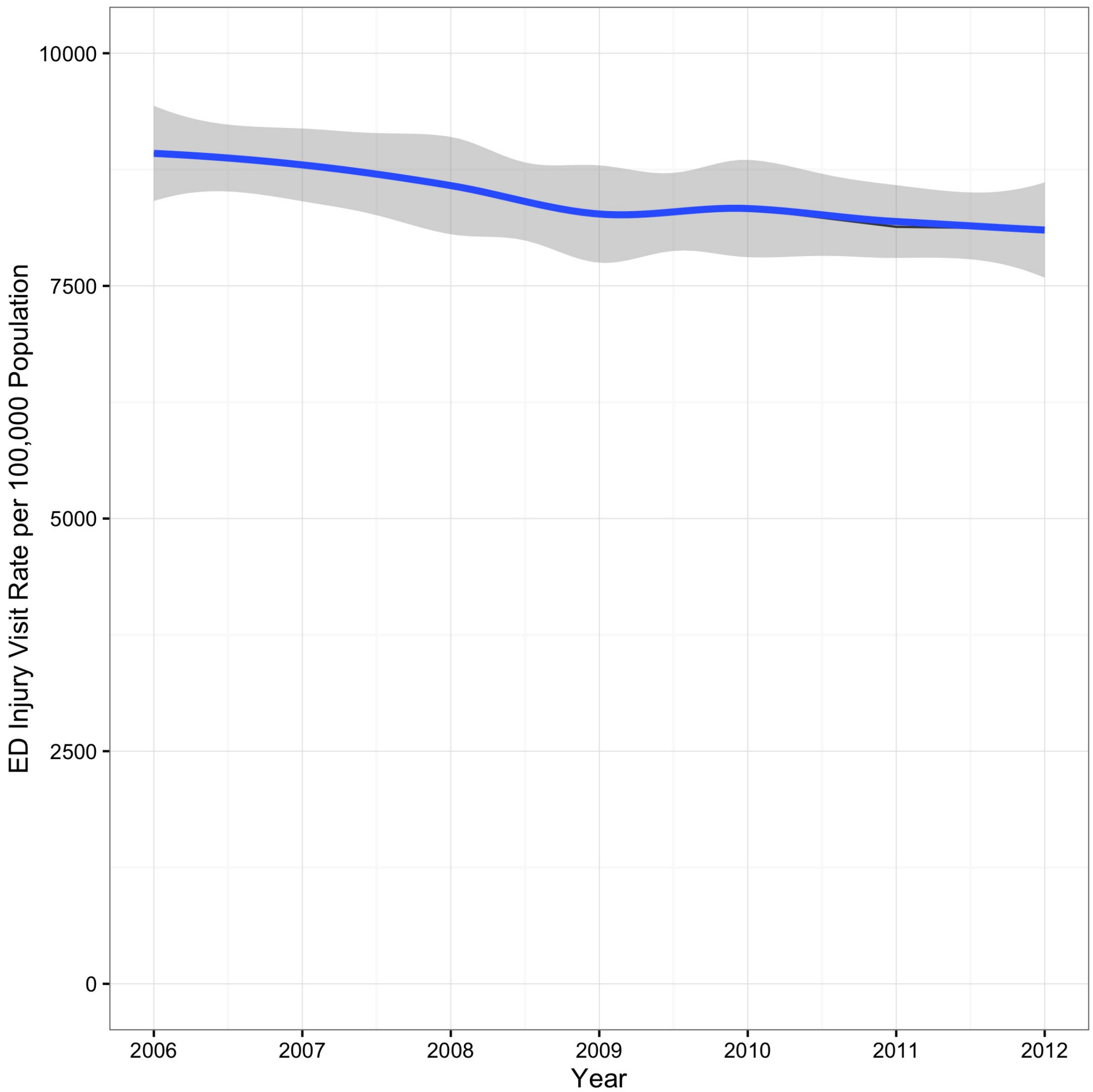


Figure 1: Emergency Department Visits for Traumatic Injury per 100,000 Population, US Hospitals, 2006-2012.

```
    2012) + geom_smooth(aes(group = 1), size = 1.5)
p3 <- p2 + ylab("ED Injury Visit Rate per 100,000 Population") +
    theme_bw() + scale_x_continuous(breaks = seq(2006, 2012,
    1))
p4 <- p3 + geom_point(aes(y = Rate, x = Year))
natRatePlot <- p4
ggsave(file = "/Users/charlie/Box Sync/hcup/NEDS/nedsNotes/natRatePlot.jpg",
    natRatePlot)
```

### 3.1.1 Age and Gender

The mean age of a person visiting an emergency department for a traumatic injury between 2006 and 2012 in the United Stats was 34.7 year old (s.e.=0.1). There was an increase in this mean age, from 33.8 years old in 2006 (s.e.=0.01) to 35.3 in 2012 (0.001). There was little change in the population-based visit rate by age group over the study period. (Table 2, Figure 2) Females represented 46.6% (s.e.=0.01) of all traumatic injury emergency department visits in the U.S. during the study period, and 34.9% (s.e.=0.01) of severely injured patients.

```
(svymean(~age, monInj, na.rm=T,multicore=T, se=T))

(svymean(~age, monInj, na.rm=T,multicore=T, se=T, byvar=~yr))

dbGetQuery(mdb, "SELECT agegrp_num, COUNT(*) AS total FROM nedsinj0612 GROUP BY agegrp_num;")

(yrCountAge1<-svytotal(~count,subset(monInj, agegrp_num==1), byvar=~yr, se=T, na.rm=T, multicore=T))
(yrCountAge2<-svytotal(~count,subset(monInj, agegrp_num==2), byvar=~yr, se=T, na.rm=T, multicore=T))
(yrCountAge3<-svytotal(~count,subset(monInj, agegrp_num==3), byvar=~yr, se=T, na.rm=T, multicore=T))
(yrCountAge4<-svytotal(~count,subset(monInj, agegrp_num==4), byvar=~yr, se=T, na.rm=T, multicore=T))
(yrCountAge5<-svytotal(~count,subset(monInj, agegrp_num==5), byvar=~yr, se=T, na.rm=T, multicore=T))

ageGroups<-c(rep("0to18",7),rep("18to45",7),rep("45to55",7),rep("65to85",7)
,rep("GT85",7))
ageCount<- c(7457903, 7379373, 7072187, 6920894, 6761431, 6845915, 6951220, 11297852,
11117346, 10860471, 10294296, 10531512, 10072285, 10024540, 4745698, 4826772, 4865721,
4808729, 5043254, 4957956, 4962435, 2318552, 2357144, 2400043, 2435519, 2485301, 2501217,
2555035, 804011.8, 838886.8, 886432.5, 916916.0, 950394.8, 969965.8, 977223.1)
ageSE <- c(5963.928, 5938.786, 5649.369, 5756.464, 5637.680, 5466.500, 5561.035, 7218.432,
       7179.338, 6877.823, 6965.587, 6952.260, 6444.134, 6604.615, 4830.582, 4872.493,
       4742.890, 4881.299, 4963.504, 4710.574, 4763.831, 3401.098, 3436.132, 3373.399,
       3520.227, 3535.614, 3394.970, 3454.130, 2013.517, 2072.277, 2064.625, 2179.210,
        2200.099, 2143.672, 2149.256)
yrs<-rep(c("2006","2007","2008","2009","2010","2011","2012"),5)
ageCountTab<-data.frame(yrs,ageGroups, ageCount, ageSE)
ageCountTab
```

```r
xtable(ageCountTab)

pops<-c(rev(c(73708179, 73902222, 74195760, 74134167, 74104602, 74019405, 73757714)),
rev(c(114166892, 113498581, 112936714, 112741499, 112594233, 112317718, 112241819)),
rev(c(82854869, 82812248, 81779634, 80272688, 78617510, 77068373, 75216272)),
rev(c(37262128, 35656696, 34904825, 34255874, 33581781, 32786166, 32298178)),
rev(c(5881617, 5712817, 5532756, 5367301, 5195840, 5039545, 4865929)))

rates<-ageCount/pops*100000
ageRateTab<-data.frame(ageGroups,yrs,rates)
ageRateTab

p1<-ggplot(data=ageRateTab, aes(x=yrs, y=rates))
p2<-p1+geom_line(aes(group=1))+ylim(0,20000)+facet_wrap(~ageGroups)
+geom_smooth(aes(group=1))
p3<-p2+ylab("ED Injury Visit Rate per 100,000 Population")+xlab("Year")+theme_bw()
ageRatePlot<-p3
ggsave(file="/Users/charlie/Box Sync/hcup/NEDS/nedsNotes/ageRatePlot.jpg",ageRatePlot)

svymean(~female, monInj, na.rm=T, multicore=T, se=T)
genderSevereTab<-svymean(~female, monInj, na.rm=T, multicore=T, se=T, byvar=~severe)
```

### 3.1.2 Deaths

During the study period, there were 73,655 (s.e.= 618) traumatic injury-related deaths in the emergency department and 229,810 (s.e.=1,078) deaths after admission for a total case fatality rate of 0.17%. The case fatality rate for all traumatic injury visits decreased from 0.18% in 2006 to 0.15% in 2012. The case-fatality rate for severely injured persons was 4.1% over the entire study period. The case-fatality rate for severely injured persons decreased from 5.2% in 2006 to 4.3% in 2012. A total of 2,155,428 observations (1.2%) were missing an entry for death.

```r
(svytotal(~died_visit, monInj, na.rm=T,multicore=T, se=T))

73655.85+229809.55
(73655.85+229809.55)/181194431*100

(svytotal(~died_visit, monInj, na.rm=T,multicore=T, se=T, byvar=~yr))

yrlyCFR<-c((11099.557+34350.103)/(25309366.032+11099.557+34350.103),
(10199.455+34637.734)/(25945095.596+10199.455+34637.734),
(11302.936+35031.686)/(25863153.970+11302.936+35031.686),
(10541.145+32439.480)/(25287032.405+10541.145+32439.480),
(10222.007+33935.553)/(25644111.014+10222.007+33935.553),
(10751.810+30232.415)/(25276147.984+10751.810+30232.415),
(9538.936+29182.576)/(25410626.436+9538.936+29182.576))
```

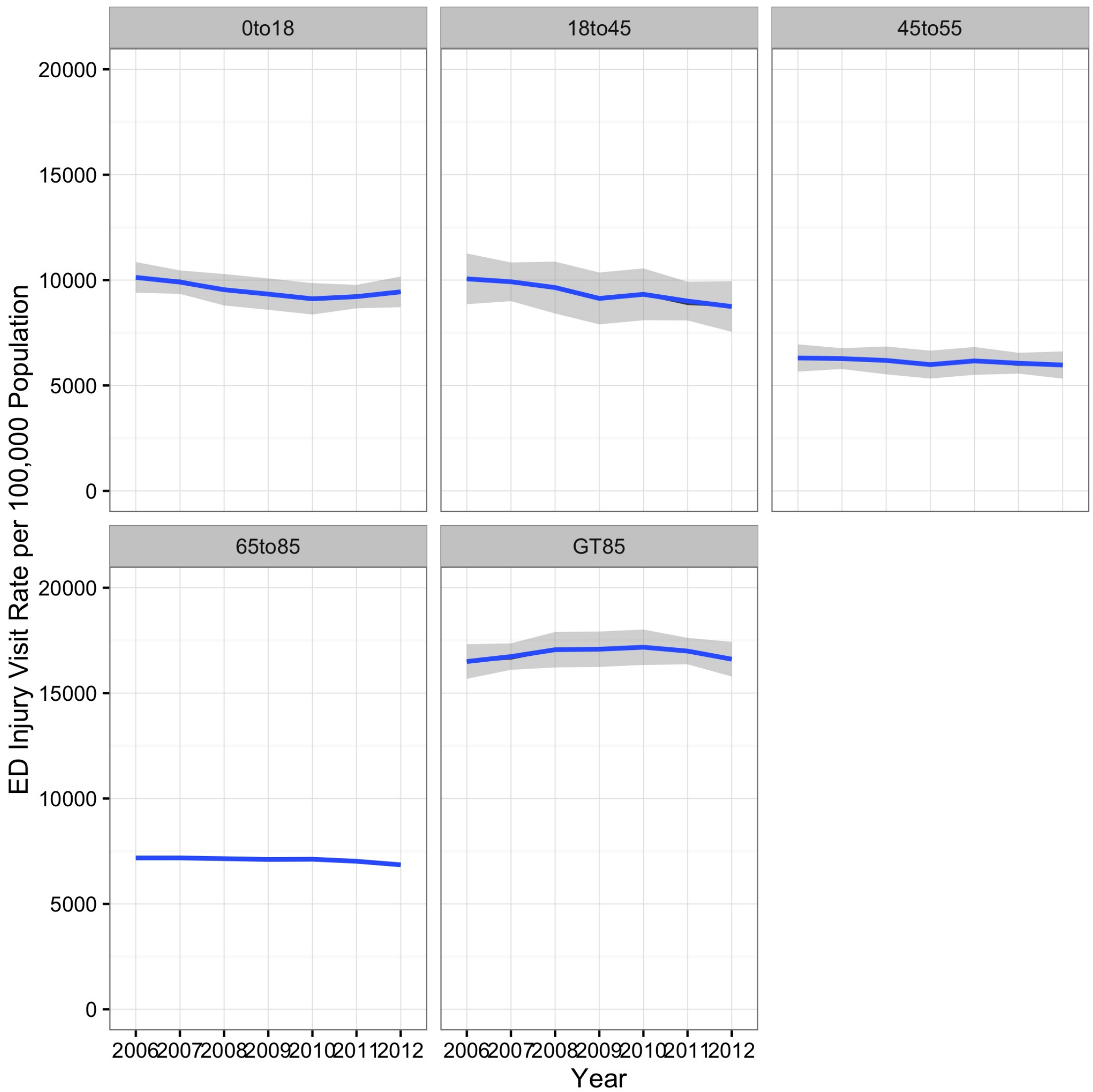


Figure 2: Emergency Department Visit Rate for Traumatic Injury per 100,000 Population by Age Group and Year, US Hospitals, 2006-2012.

```
yrlyCFR*100

(severeDeaths<-svytotal(~died_visit,design=subset(monInj, severe==1 ),
se=T, multicore=T))
(43223.12+141311.71)/(3656761.56+43223.12+141311.71)*100

(severeDeathsYr<-svytotal(~died_visit,design=subset(monInj, severe==1 ),
se=T, byvar=~yr,  multicore=T))
severeYrlyCFR<-c((6881.5440+20680.2415)/(498559.1193+6881.5440+20680.2415),
(5666.1956+20836.9330)/(519545.1277+5666.1956+20836.9330),
(6835.5266+21923.4476)/(545772.3793+6835.5266+21923.4476),
(6275.5921+19335.7035)/(498611.2189+6275.5921+19335.7035),
(5916.9545+21676.6512)/(545927.7387+5916.9545+21676.6512),
(5828.7366+18557.6807)/(508514.4682+5828.7366+18557.6807),

severeYrlyCFR*100

(missing<-sum(1269201,529588,175366,46341,83623,30205,21104))
missing/181194431*100
```

## 3.2 Injury Severity and Comorbidities

There was some graphical evidence of a decline in injury severity in the final two years of the study period. (Figure 3)

```
# NB - use dplyr connection to external sql database to get quantiles
res <- nedsinj %>% group_by (agegrp,yr) %>% do(q=data.frame(quantile(.$iciss)))
saveRDS(res, file="/Users/charlie/Box Sync/hcup/NEDS/nedsNotes/icissQuant.rds")
icissQuant<-readRDS("/Users/charlie/Box Sync/hcup/NEDS/nedsNotes/icissQuant.rds")
icissQuant<-data.frame(icissQuant)

# extract IQR numbers to variables
X0.25<-rep(NA,35)
for (i in 1:35){
        X0.25[i]<-icissQuant$q[[i]][2,1]
}
X0.25

X0.5<-rep(NA,35)
for (i in 1:35){
        X0.5[i]<-icissQuant$q[[i]][3,1]
}
X0.5

X0.75<-rep(NA,35)
for (i in 1:35){
```

```r
        X0.75[i]<-icissQuant$q[[i]][4,1]
}
X0.75

icissQuant$X0.25<-X0.25
icissQuant$X0.5<-X0.5
icissQuant$X0.75<-X0.75

names(icissQuant)

mf_labeller <- function(var, value){
    value <- as.character(value)
    if (var=="agegrp") {
        value[value==1]<-"0 to 18"
        value[value==2]<-"18 to 44"
        value[value==3]<-"45 to 64"
        value[value==4]<-"65 to 84"
        value[value==5]<-"Over 84"
    }
    return(value)
}

icissQuant$ageGrp.fact<-factor(icissQuant$agegrp, labels=c("0 to 18","18 to 44","45 to 64","65 to 84"

p1<-ggplot(data=icissQuant, aes(y=X0.5, ymin=X0.25, ymax=X0.75, x=yr))+ylim(0,1)+xlab("Year")+ylab("
p2<-p1+geom_linerange(alpha = I(12/12))
p3<-p2+geom_point(aes(y=X0.5, x=yr))
p4<-p3+facet_grid(.~ageGrp.fact)+theme_bw()
p4
ageICISSPlot_yr_A<-p4

p1<-ggplot(data=icissQuant, aes(y=X0.5, ymin=X0.25, ymax=X0.75, x=yr))+ylim(.97,1)+xlab("Year")+ylab
p2<-p1+geom_linerange(alpha = I(12/12))
p3<-p2+geom_point(aes(y=X0.5, x=yr))
p4<-p3+facet_grid(.~ageGrp.fact)+theme_bw()
p4
ageICISSPlot_yr_B<-p4

ageICISSPlot_yr_AB<-plot_grid(ageICISSPlot_yr_A, ageICISSPlot_yr_B,  labels=c('A', 'B'),ncol = 1, re
ageICISSPlot_yr_AB
ggsave(file="/Users/charlie/Box Sync/hcup/NEDS/nedsNotes/ageICISSPlot_yr_AB.jpg",ageICISSPlot_yr_AB
```

The proportion of visits with a Charlson comorbidity index score of 2 or greater nearly doubled from 0.67% (sd=0.004%) of all visits in2006 to 1.12% (sd=0.005%) of all visits in 2012. During the study period, 0.90% (sd=0.002%) of individuals who died in the emergency department due to traumatic injury had a Charlson comorbidity index score of 2 or greater. By contrast, 20.9% of individuals who died after admission due to traumatic injury had a

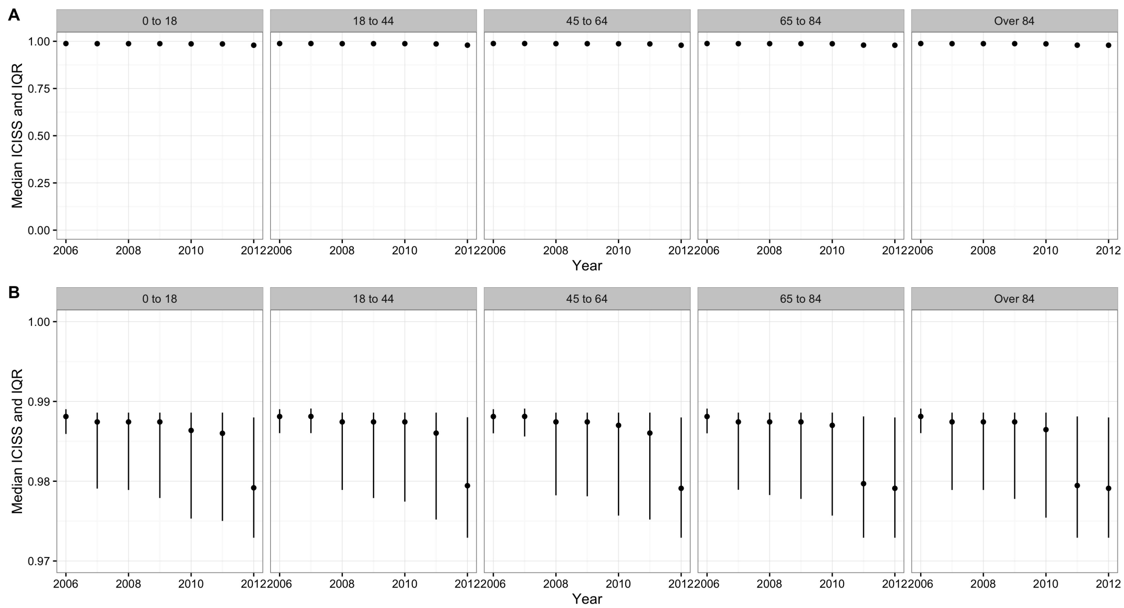


Figure 3: Median and Intraquartile Range. ICD9 Derived Injury Severity Scores by Age Group and Year, total scale (A) and focused scale (B). Emergency Department Visits for Traumatic Injury, US Hospitals, 2006-2012.

Charlson comorbidity index score of 2 or greater.

```
# charlson by year
(yrCharlson<-svytotal(~charleson_severe, byvar=~yr, se=T, na.rm=T, multicore=T, design=monInj))

num <- c(179001.8, 200497.5, 220599.6, 240384.5, 256399.1, 279541.6, 285049.1)
denom <- c(26624017, 26519522, 26084855, 25376355, 25771892, 25347338, 25470453)

num/denom*100

num.2006<-rnorm(1000, mean=179001.8, sd=958.8213)
denom.2006<-rnorm(1000, mean=26624017, sd=10381.190)
prop.2006<-num.2006/denom.2006
mean(prop.2006)*100 # 0.6723811
sd(prop.2006)*100 # 0.003697841

num.2012<-rnorm(1000, mean=285049.1, sd=1160.2356)
denom.2012<-rnorm(1000, mean=25470453, sd=9400.411)
prop.2012<-num.2012/denom.2012
mean(prop.2012)*100 # 1.119029
sd(prop.2012)*100 # 0.004728594


# charlson by mortality
(diedCharlson<-svytotal(~charleson_severe, byvar=~died_visit, se=T, na.rm=T, multicore=T, design=mon
```

```
# charlson didn't die
1604507.8176/178735533.44*100 # 0.8976994
severe0.died<-rnorm(1000,mean=1604507.8176,sd=2802.40350)
died0<-rnorm(1000,mean=178735533.44,sd=5366.8634)
ratio.died0<-(severe0.died/died0)
mean(ratio.died0)*100 # 0.8976256
sd(ratio.died0)*100 # 0.001561414

# charlson severe died in ED
863.3088/73655.85*100 # 1.172084
severe1.died<-rnorm(1000,mean=863.3088,sd=70.26793)
died1<-rnorm(1000,mean=73655.85,sd=617.5398)
ratio.died1<-(severe1.died/died1)
mean(ratio.died1)*100 # 1.177934
sd(ratio.died1)*100 # 0.09360673



# charlson severe died in hospital
47915.0451/229809.55*100 # 20.84989
severe2.died<-rnorm(1000,mean=47915.0451,sd=489.66054)
died2<-rnorm(1000,mean=229809.55,sd=1077.7435)
ratio.died2<-(severe2.died/died2)
mean(ratio.died2)*100 # 20.84989
sd(ratio.died2)*100 # 0.2420907
```

## 3.3 Hospital Type

During the study period, there were 79,632,500 (s.e.= 2,154) traumatic injury-related emergency department visits to non-teaching hospitals in metropolitan areas, and 64,792,147 (s.e.=3,005) visits to teaching hospitals in metropolitan areas. A remaining 36,486,771 (s.e.=2,064) visits were to non-metropolitan hospitals. The risk ratio for the number of traumatic injury emergency department visits characterized as severe presenting to metropolitan teaching compared to non-teaching metropolitan hospitals was 2.16 (95% CI 2.01, 2.30). There were 1,036,897 (s.e.= 2,176) total traumatic injury fatalities at metropolitan teaching hospital emergency departments during the study period, compared to 1,002,419 (s.e. = 2,115) at non-teaching metropolitan emergency departments, for a risk ratio of 1.27 (95% CI 1.15, 1.39).

```
(hosp1<-svytotal(~hosp_ur_teach, se=T, na.rm=T, multicore=T, design=monInj))

severe.teach<-rnorm(1000, mean=2133919.7, sd=3198.934)
denom.teach<-rnorm(1000, mean=64792147, sd=3005.304)
severe.nonteach<-rnorm(1000, mean=1214208.0, sd=2418.825)
denom.nonteach<-rnorm(1000, mean=79632500, sd=2153.931)
```

```
point<-mean((severe.teach/denom.teach)/(severe.nonteach/denom.nonteach)) # 2.160192
s.e.<-sd((severe.teach/denom.teach)/(severe.nonteach/denom.nonteach)) # 0.005294831

point+(1.96*sqrt(s.e.)) # 2.302813
point-(1.96*sqrt(s.e.)) # 2.017572

(hosp4<-svytotal(~hosp_ur_teach, se=T, na.rm=T, byvar=~died, multicore=T, design=monInj))

died.teach<-rnorm(1000, mean=1036897.4, sd=2176.372)
denom.teach<-rnorm(1000, mean=64792147, sd=3005.304)
died.nonteach<-rnorm(1000, mean=1002419.3, sd=2115.306)
denom.nonteach<-rnorm(1000, mean=79632500, sd=2153.931)

point<-mean((died.teach/denom.teach)/(died.nonteach/denom.nonteach)) # 1.271478
s.e.<-sd((died.teach/denom.teach)/(died.nonteach/denom.nonteach)) # 0.003578975

point+(1.96*sqrt(s.e.)) # 1.388734
point-(1.96*sqrt(s.e.)) # 1.154222
```

There were 33,332,579 (s.e.=1,765) traumatic injury related emergency department visits to level 1 or level 2 trauma centers during the study period, of which 1,603,010 (s.e.=2,779) were classified as severe for a proportion of 4.8% (95% CI 4.6, 4.9). By contrast, the proportion of visits to non-level 1 or 2 trauma centers was 1.3% (95% CI 1.2, 1.4) The risk ratio for severely injured persons presenting to level 1 or 2 trauma centers was 3.8 (95% CI 3.6, 4.0). There were a total of 511,021 (s.e.=1599) deaths in level 1 or 2 trauma centers for a case fatality ratio of 1.5% (95% CI 1.4, 1.6). There were a total of 1,857,446 (s.e.=2987) traumatic-injury deaths in non-level 1 or 2 trauma centers for a case fatality ratio of 1.3% (95% CI 1.2, 1.4). The risk ratio for fatality in level 1 or 2 trauma centers vs. non-level 1 or 2 trauma centers was 1.1 (95% CI 0.9, 1.2). In a survey-adjusted logistic regression for injury-related mortality controlling for age, gender, injury severity and comorbidity, there was a non-statistically significant benefit to treatment at a level 1 or level 2 trauma center. (Table 3)

```
# injury severity
(traum1<-svytotal(~trauma_center, se=T, na.rm=T, multicore=T, design=monInj))

hosp_trauma1<- rnorm(1000, mean=19387024, sd=1050.1548)
hosp_trauma2<- rnorm(1000, mean=13945556, sd=1503.0178)

trauma<-hosp_trauma1+hosp_trauma2
mean(trauma) # 33332579
sd(trauma) # 1765.491

hosp_trauma1_1<-rnorm(1000, mean=1137574.2, sd= 2262.1580)
hosp_trauma2_1<-rnorm(1000, mean= 465434.9, sd= 1565.1410)
```

```r
severe.trauma<-hosp_trauma1_1+hosp_trauma2_1
mean(severe.trauma)# 1603010
sd(severe.trauma) # 2779.387

severe.trauma.prop<-severe.trauma/trauma
point<-mean(severe.trauma.prop)*100
s.e.<-sd(severe.trauma.prop)*100

point+(1.96*sqrt(s.e.))
point-(1.96*sqrt(s.e.))

(traum2<-svytotal(~hosp_trauma, se=T, na.rm=T, byvar=~severe, multicore=T, design=monInj))

hosp_trauma0  <-rnorm(1000,mean= 103442645   , sd=2711.6200)
hosp_trauma3  <-rnorm(1000,mean=  13574806   , sd=2059.2076)
hosp_trauma4  <-rnorm(1000,mean=  16959232   , sd= 663.5463)


hosp_trauma0_1<-rnorm(1000,mean=   1224890.9 , sd=2353.2819)
hosp_trauma3_1<-rnorm(1000,mean=    252722.6 , sd=1407.5726)
hosp_trauma4_1<-rnorm(1000,mean=    224087.0 , sd= 973.1917)


nontrauma <- hosp_trauma0+ hosp_trauma3+ hosp_trauma4
severe.nontrauma <- hosp_trauma0_1+ hosp_trauma3_1+ hosp_trauma4_1

severe.nontrauma.prop<-severe.nontrauma/nontrauma
(point<-mean(severe.nontrauma.prop)*100) # 1.270078
(s.e.<-sd(severe.nontrauma.prop)*100) # 0.002206135

point+(1.96*sqrt(s.e.))
point-(1.96*sqrt(s.e.))




# mortality
(traum4<-svytotal(~hosp_trauma, se=T, na.rm=T, byvar=~died, multicore=T, design=monInj))


hosp_trauma1_1<-rnorm(1000, mean=    309604.49, sd= 1200.9366)
hosp_trauma2_1<-rnorm(1000, mean=    201353.46, sd= 1089.5077)
died.trauma<-hosp_trauma1_1+hosp_trauma2_1
mean(died.trauma)
sd(died.trauma)

hosp_trauma1<- rnorm(1000, mean=19387024, sd=1050.1548)
hosp_trauma2<- rnorm(1000, mean=13945556, sd=1503.0178)
```

```r
trauma<-hosp_trauma1+hosp_trauma2


died.trauma.prop<-died.trauma/trauma
(point<-mean(died.trauma.prop)*100)
(s.e.<-sd(died.trauma.prop)*100)

point+(1.96*sqrt(s.e.))
point-(1.96*sqrt(s.e.))

hosp_trauma0_1<-rnorm(1000,   1536670.23, 2666.5217)
hosp_trauma3_1<-rnorm(1000,    300904.24, 1135.9552)
hosp_trauma4_1<-rnorm(1000,     19876.10,  293.4770)

died.nontrauma<-hosp_trauma0_1+hosp_trauma3_1+hosp_trauma4_1
mean(died.nontrauma) # 1857446
sd(died.nontrauma) # 2987.042

hosp_trauma0  <-rnorm(1000,mean= 103442645   , sd=2711.6200)
hosp_trauma3  <-rnorm(1000,mean=  13574806   , sd=2059.2076)
hosp_trauma4  <-rnorm(1000,mean=  16959232   , sd= 663.5463)
nontrauma <- hosp_trauma0+ hosp_trauma3+ hosp_trauma4

died.nontrauma.prop<-died.nontrauma/nontrauma
(point<-mean(severe.nontrauma.prop)*100) # 1.270078
(s.e.<-sd(severe.nontrauma.prop)*100) # 0.002206135

point+(1.96*sqrt(s.e.)) # 1.362139
point-(1.96*sqrt(s.e.)) # 1.178018

rr.traumaCenter<-(died.trauma/trauma)/(died.nontrauma/nontrauma)
point<-mean(rr.traumaCenter) # 1.105819
s.e.<-sd(rr.traumaCenter) # 0.003881173
point+(1.96*sqrt(s.e.)) # 1.227925
point-(1.96*sqrt(s.e.))# 0.9837126
```

```r
d<-datadist(inj)
options( datadist = "d" )

fatalOR2<-lrm(died ~ ageGrp.num + FEMALE + teach + traumaCenter + severe + Charlson, data=inj, x=T, y=
fatalOR2robust<-robcov(fatalOR2, cluster=inj$NEDS_STRATUM)
summary(fatalOR2robust)
```

## 3.4 Injury Categories and Causes

Table 4 lists the most common Barell Matrix locations of traumatic injuries presenting to US emergency departments between 2006 and 2012. Population-based rates of Barell matrix injury categories differed by age group over time. ?? Overall rates were highest for the over age 84 group. The most notable changes were seen at the extremes of ages. The rate of traumatic brain injuries for the over age 84 group increased 29.5% over the study period, from 545.0/100,000 in 2006 to 705.8/100,000 in 2012. At the same time, lower extremity injuries in this age group, including hip fractures, decreased 12.3%, from 5,108/100,000 in 2006 to 4,477/100,000 in 2012. The under age 18 group experienced a 44.9% increase in traumatic brain injury diagnoses, from 190.1/100,000 in 2006 to 275.4/100,000 in 2012. Over that time, the under age 18 group also experience a 14.7% decline in injuries to the torso, from 352.3/100,000 in 2006 to 300.7/100,000 in 2012.

```
(ISRCODE.descr<-svytotal(~isrcode_descr, se=T,  multicore=T, design=monInj))

# barell by age and year
library(freqweights)
c<-tablefreq(inj, vars=c("ISRSITE2.descr","ageGrp.num","YEAR"), freq="DISCWT")
as.data.frame(c)
saveRDS(as.data.frame(c), file="/Users/charlie/Box Sync/hcup/NEDS/nedsNotes/isrSiteXyrXage.rds")
isrSiteXyrXage<-readRDS("/Users/charlie/Box Sync/hcup/NEDS/nedsNotes/isrSiteXyrXage.rds")

isrSiteXyrXage<-isrSiteXyrXage[!is.na(isrSiteXyrXage$ISRSITE2.descr),] # remove NA's

# add population figures, calculate rates
pops<-c(rev(c(73708179, 73902222, 74195760, 74134167, 74104602, 74019405, 73757714)),
rev(c(114166892, 113498581, 112936714, 112741499, 112594233, 112317718, 112241819)),
rev(c(82854869, 82812248, 81779634, 80272688, 78617510, 77068373, 75216272)),
rev(c(37262128, 35656696, 34904825, 34255874, 33581781, 32786166, 32298178)),
rev(c(5881617, 5712817, 5532756, 5367301, 5195840, 5039545, 4865929)))

pops2<-rep(pops,9)
isrSiteXyrXage$pop<-pops2
isrSiteXyrXage$rates<-isrSiteXyrXage$freq/isrSiteXyrXage$pop*100000

mf_labeller <- function(var, value){
    value <- as.character(value)
    if (var=="ageGrp.num") {
        value[value==1]<-"0 to 18"
        value[value==2]<-"18 to 44"
        value[value==3]<-"45 to 64"
        value[value==4]<-"65 to 84"
        value[value==5]<-"Over 84"
    }
    return(value)
}
```

```
p1<-ggplot(data=isrSiteXyrXage[(isrSiteXyrXage$ISRSITE2.descr=="TORSO" | isrSiteXyrXage$ISRSITE2.de
plot1<-p1+geom_line()+facet_wrap(~ageGrp.fact)+geom_smooth()+theme_bw()+scale_color_manual(values=c

p1<-ggplot(data=isrSiteXyrXage[(isrSiteXyrXage$ISRSITE2.descr=="TORSO" | isrSiteXyrXage$ISRSITE2.de
plot2<-p1+geom_line()+facet_wrap(~ageGrp.fact)+geom_smooth()+theme_bw()+scale_color_manual(values=c

plot_grid(plot1, plot2, align='h')

ggsave(file="/Users/charlie/Box Sync/hcup/NEDS/nedsNotes/ageYrBarellPlot.jpg",ageYrBarellPlot)


isrSiteXyrXage

((705.7993522-544.9629206)/544.9629206)*100 # oldest-tbi, 29.51328
(5108.0763901-4477.9036705)/5108.0763901*100 # oldest-low extremity, 12.33679


(275.4112413-190.1029154)/190.1029154*100 # youngest, tbi, 44.87481
(352.3085770-300.6776326)/352.3085770*100 # youngest, torso, 14.65503
```

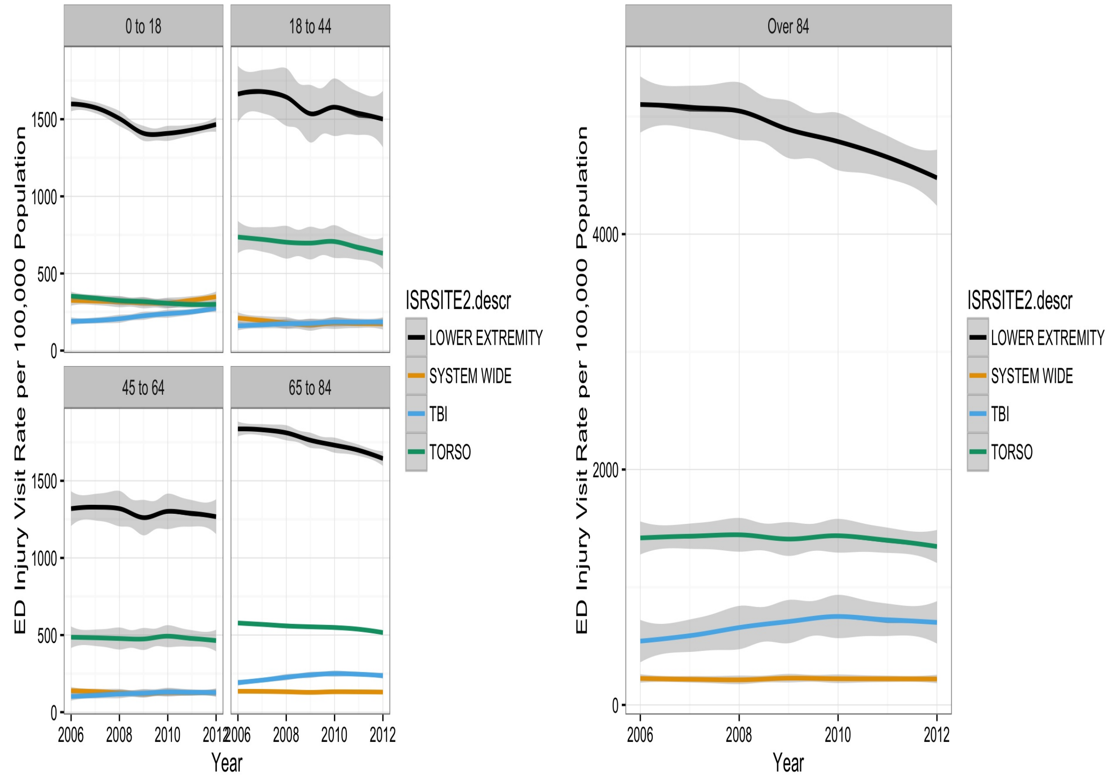


Figure 4: Barell Matrix Injury Types by Age Group and Year. Emergency Department Visits for Traumatic Injury, US Hospitals, 2006-2012.

Emergency department visits due to motor vehicles and other transportation declined 3.5%

from 2,656,790 (s.e.=3,652) or 866.0/100,000 total US population in 2009 to 2,623,246 (s.e. = 3,568) or 835.8/100,000 population. In the youngest age group, children younger than 18 years old, the motor vehicle and transportation related injury rate declined 8.3%, from 480.2/100,000 population to 440.4/100,000 population. Total injuries due to fall declined 1.4%, from 7,713,447 (s.e.=6098) visits or 2,514.4/100,000 total population in 2009 to 7,781,310 (s.e.= 5,839) or 2479.1/100,000 in 2012. The emergency department visit rate for falls among persons older than 65 remained essentially unchanged, from 7,713,447 (s.e.=6,098) or 5.3/100,000 in 2009 to 7,781,310 (s.e. = 5839) or 5.1/100,000 population in 2012. There were a total of 71,111 (s.e.=613) firearm-related emergency department visits in 2009 for a population-based rate of 23.2/100,000. This increased 3.9% to 75,559 (s.e.= 610) in 2012 for a rate of 24.1/100,000. A larger increase was seen in the 18 to 44 year old age group, where there were 52,187 (s.e.=527) firearm-related emergency department visits in 2009 or a rate of 46.3/100,000 , which increased 7.2% to 56,644 (s.e.=528) or a rate of 49.6/100,000. In 2006, there were 351 (s.e.= 46) emergency department visits for firearm-related injuries to children 5 years old and younger this increased 29.3% to 450 (s.e. = 49) emergency department visits for children 5 years and younger for firearm-related injuries in 2012. (Figure 5) Additional causes of injury are listed in table 7

```
# mvc
(mvc<-svytotal(~count,design=monInj, se=T, byvar=~yr+injury_mvt,  multicore=T))
2656790/(306771529)*100000 # 866.0484, 2009
2623246/(313873685 )*100000 # 835.7649, 2012
(866.0484-835.7649)/866.0484*100

(mvc.kids<-svytotal(~count,design=subset(monInj, agegrp_num==1 ), se=T, byvar=~yr+injury_mvt,  multic
356012.4/74134167*100000 # 2009 kids 480.2272
324605.9/73708179*100000 # 2012 kids 440.3933
(480.2272-440.3933)/480.2272*100



# falls
(fall<-svytotal(~count,design=monInj, se=T, byvar=~yr+injury_fall,  multicore=T))
7713447/(306771529)*100000 # 2514.395
7781310/(313873685 )*100000 # 2479.121
(2514.395-2479.121)/2514.395*100

(fall<-svytotal(~count,design=subset(monInj, agegrp_num==4 | agegrp_num==5 ), se=T, byvar=~yr+injury_
2104056/(34255874+5367301)*100 # 5.310165 2009
2220069/(37262128+5881617)*100 # 5.145749 2012

# firearms
(firearm<-svytotal(~count,design=monInj, se=T, byvar=~yr+injury_firearm))
71110.60/(306771529)*100000 # 2009 all 23.18031
75559.37/(313873685 )*100000 # 2012 all 24.07318
(24.07318-23.18031)/23.18031*100

(gunsKids<-svytotal(~count,design=subset(monInj,agegrp_num==2 ), se=T, byvar=~yr+injury_firearm,  mu
```

```
52187.19/112741499*100000 # 46.28925
56644.07/114166892*100000 # 49.61515
(49.61515-46.28925)/46.28925*100


(gunsKids2<-svytotal(~count,design=subset(monInj, age < 6 ), se=T, byvar=~yr+injury_firearm,  multico
350.6764/sum(4003587, 4078797, 4103002, 4025675, 4033457, 4070265)*100000 # 1.442235
449.9123/sum(3941616, 3976214, 3978498, 3981693, 4111675, 4131791)*100000 # 1.865193
(1.865193-1.442235)/1.442235*100 # 29.32657


kid.shot<-data.frame(Frequency = c(350.6764, 379.5269, 440.4196, 449.9123),
se = c(46.16911, 42.25565, 46.94518, 48.64465),
Population=c(sum(4003587, 4078797, 4103002, 4025675, 4033457, 4070265),
sum(3962091, 3965687, 3970771, 4101830, 4121819, 4087540),
sum(3951196, 3957772, 4090760, 4111845, 4077502, 4064628),
sum(3941616, 3976214, 3978498, 3981693, 4111675, 4131791)))

kid.shot$Rate<-kid.shot$Frequency/kid.shot$Population*100000
kid.shot$Rate.U<-(kid.shot$Frequency+kid.shot$se)/kid.shot$Population*100000
kid.shot$Rate.L<-(kid.shot$Frequency-kid.shot$se)/kid.shot$Population*100000
kid.shot$Year<-2009:2012

p1<-ggplot(data=kid.shot, aes(y=Rate, ymin=Rate.L, ymax=Rate.U, x=Year))+xlab("Year")+ylab("Rate per
p2<-p1+geom_linerange(alpha = I(12/12))
p3<-p2+geom_point(aes(y=Rate, x=Year))+ylim(0,2.5)
p3+theme_bw()
kidShotPlot<-p3+theme_bw()

ggsave(file="/Users/charlie/Box Sync/hcup/NEDS/nedsNotes/kidShotPlot.jpg",kidShotPlot)

ccs1<-svytotal(~count,design=monInj, se=T, byvar=~e_ccs1)
```

## 3.5 Costs

The total inflation-adjusted costs of emergency department injury care in the United Staets between 2006 and 2012 was $99.75 Billion (s.e.=26,685,092). The average cost for an traumatic injury-related emergency department visit was $645 (s.e.=0.17). This cost increased every year. (Figure 6)

```
print(svytotal(~adj_cost, svydes, na.rm=T))
9.975e+10/1000000000
system.time(meanCOST<-svymean(~adj_cost, monInj, se=T, na.rm=T, multicore=T))


COSTbyYear<-svyby(~adj_cost, ~yr, svydes, svymean, na.rm=T, vartype = c( 'ci' , 'se' ))
```

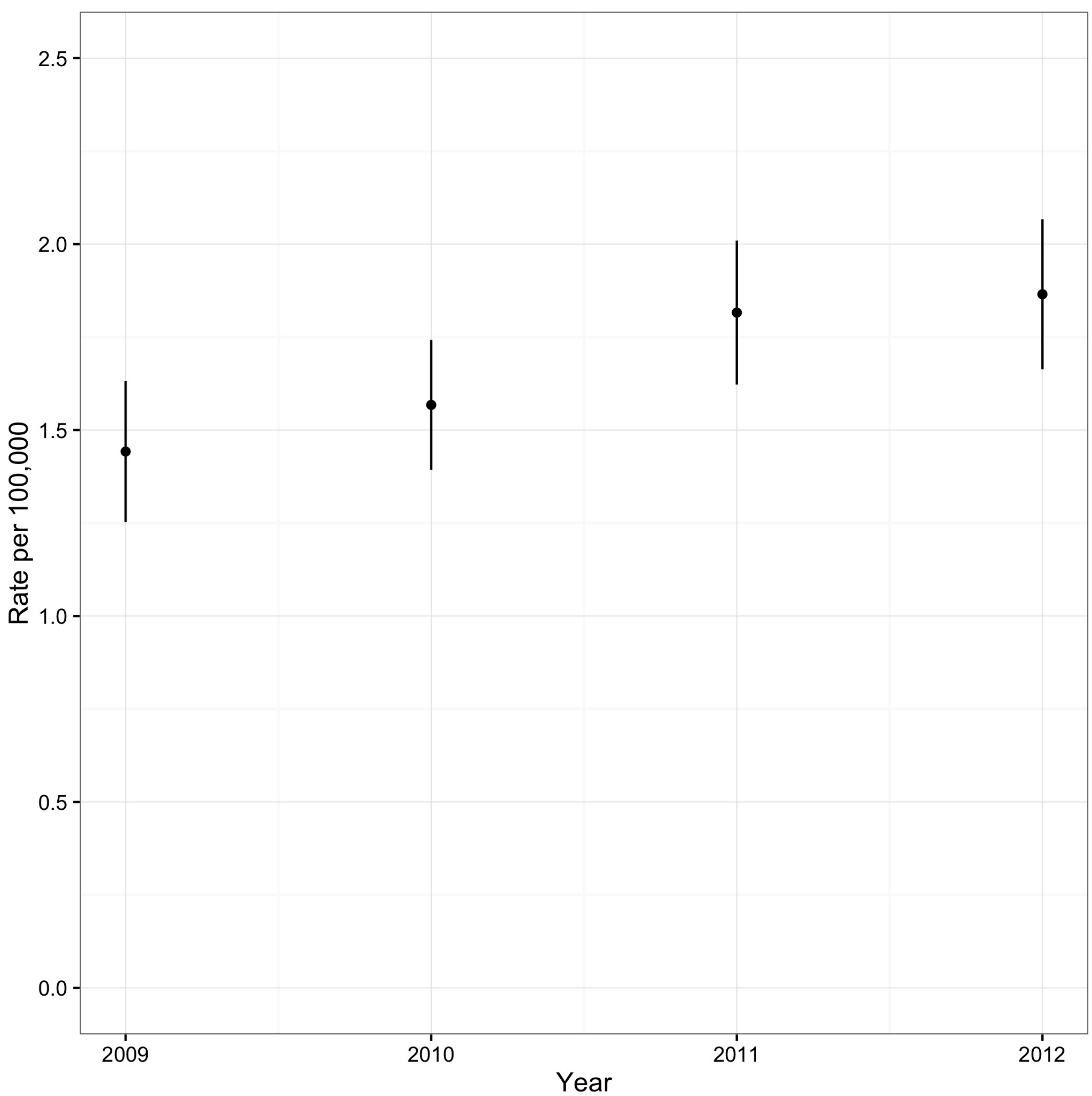


Figure 5: Rate per 100,000 Population, Emergency Department Visits for Firearm Related Injuries in Children 5 Years and Younger. United States, 2009-2012.

```r
system.time(COSTbyYear<-svyby(~adj_cost, ~yr, svydes, svymean, na.rm=T, vartype = c( 'ci' , 'se' )))
cost.dat<-data.frame(Cost= c(424.593, 495.4825, 580.3209, 638.8913, 713.5966, 786.0859, 873.0273),
Cost.L=c(424.03, 494.82, 579.56, 638.04, 712.69, 785.02, 871.84),
Cost.U=c(424.16, 496.15, 581.09, 639.75, 714.51, 787.15, 874.21),
Year=2006:2012)


p1<-ggplot(data=cost.dat, aes(y=Cost, ymin=Cost.L, ymax=Cost.U, x=Year))+xlab("Year")+ylab("Average
p2<-p1+geom_linerange(alpha = I(12/12))
p3<-p2+geom_point(aes(y=Cost, x=Year))+ylim(0,1000)
p3+theme_bw()+geom_smooth()
avgCostPlot<-p3+theme_bw()+geom_smooth()
ggsave(file="/Users/charlie/Box Sync/hcup/NEDS/nedsNotes/avgCostPlot.jpg",avgCostPlot)

# most frequent
dxs<-as.data.frame(sort(table(inj$dx1), decreasing=T)[1:10])
dxCost<-svyby(~adj_cost, ~dx1, subset(svydes2, dx1 %in% rownames(dxs)), svymean, na.rm=T, vartype = c
dxTotCosts<-svyby(~adj_cost, ~dx1, subset(svydes2, dx1 %in% rownames(dxs), select= c("adj_cost", "dx1

# most expensive
expensiveDxs<-as.data.frame(sort(tapply(inj$adj_cost,inj$dx1,mean),  decreasing=T))[1:10,]
avgExpensDxs<-svyby(~adj_cost, ~dx1, subset(svydes2, dx1 %in% rownames(expensiveDxs), select= c("adj_
totExpensDxs<-svyby(~adj_cost, ~dx1, subset(svydes2, dx1 %in% rownames(expensiveDxs), select= c("adj_
```

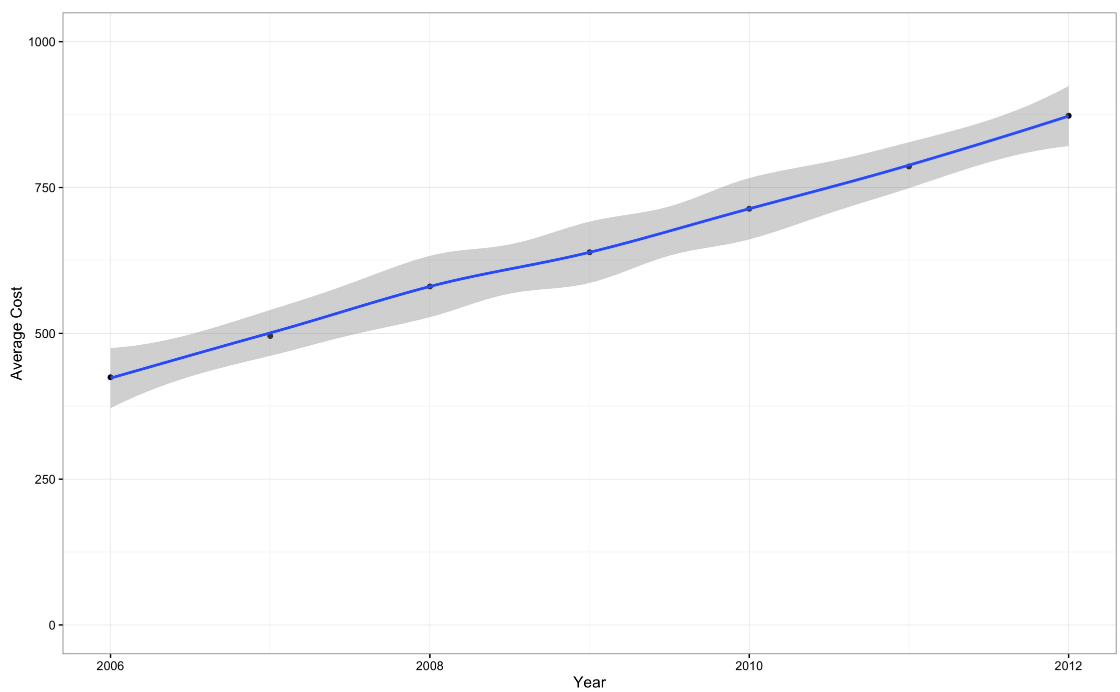


Figure 6: Average Cost of Injury-Related Emergency Department Visit by Year, 2012 Inflation-Adjusted Dollars. United States Hospitals, 2006-2012.

# 4 File Creation

## 4.1 Read csv's into R

Use data.table::fread() to quickly read in the large NEDS .csv files. [4]. After reading in a file, check against summary statistics for that year available from AHRQNet or pdfs.

```
neds.2006.core <- fread("~/2006/NEDS_2006_Core.csv")
names(neds.2006.core) <- c("AGE", "AMONTH", "AWEEKEND", "CHRON1",
    "CHRON2", "CHRON3", "CHRON4", "CHRON5", "CHRON6", "CHRON7",
    "CHRON8", "CHRON9", "CHRON10", "CHRON11", "CHRON12", "CHRON13",
    "CHRON14", "CHRON15", "DIED_VISIT", "DISCWT", "DISP_ED",
    "DQTR", "DX1", "DX2", "DX3", "DX4", "DX5", "DX6", "DX7",
    "DX8", "DX9", "DX10", "DX11", "DX12", "DX13", "DX14", "DX15",
    "DXCCS1", "DXCCS2", "DXCCS3", "DXCCS4", "DXCCS5", "DXCCS6",
    "DXCCS7", "DXCCS8", "DXCCS9", "DXCCS10", "DXCCS11", "DXCCS12",
    "DXCCS13", "DXCCS14", "DXCCS15", "ECODE1", "ECODE2", "ECODE3",
    "ECODE4", "EDEVENT", "E_CCS1", "E_CCS2", "E_CCS3", "E_CCS4",
    "FEMALE", "HCUPFILE", "HOSP_ED", "HOSP_REGION", "INTENT_SELF_HARM",
    "KEY_ED", "NDX", "NECODE", "NEDS_STRATUM", "PAY1", "PAY2",
    "PL_NCHS2006", "TOTCHG_ED", "YEAR", "ZIPINC_QRTL")
saveRDS(neds.2006.core, "~/2006/NEDS_2006_Core.rds")
nis.2006.core <- readRDS("~/2006/NEDS_2006_Core.rds")


neds.2007.core <- fread("~/NEDS_2007_Core.csv")
names(neds.2007.core) <- c("AGE", "AMONTH", "AWEEKEND", "CHRON1",
    "CHRON2", "CHRON3", "CHRON4", "CHRON5", "CHRON6", "CHRON7",
    "CHRON8", "CHRON9", "CHRON10", "CHRON11", "CHRON12", "CHRON13",
    "CHRON14", "CHRON15", "DIED_VISIT", "DISCWT", "DISP_ED",
    "DQTR", "DX1", "DX2", "DX3", "DX4", "DX5", "DX6", "DX7",
    "DX8", "DX9", "DX10", "DX11", "DX12", "DX13", "DX14", "DX15",
    "DXCCS1", "DXCCS2", "DXCCS3", "DXCCS4", "DXCCS5", "DXCCS6",
    "DXCCS7", "DXCCS8", "DXCCS9", "DXCCS10", "DXCCS11", "DXCCS12",
    "DXCCS13", "DXCCS14", "DXCCS15", "ECODE1", "ECODE2", "ECODE3",
    "ECODE4", "EDEVENT", "E_CCS1", "E_CCS2", "E_CCS3", "E_CCS4",
    "FEMALE", "HCUPFILE", "HOSP_ED", "HOSP_REGION", "INTENT_SELF_HARM",
    "KEY_ED", "NDX", "NECODE", "NEDS_STRATUM", "PAY1", "PAY2",
    "PL_NCHS2006", "TOTCHG_ED", "YEAR", "ZIPINC_QRTL")
saveRDS(neds.2007.core, "~/2007/NEDS_2007_Core.rds")
nis.2007.core <- readRDS("~/2007/NEDS_2007_Core.rds")


neds.2008.core <- fread("~/NEDS_2008_Core.csv")
```

[4]Note: Takes about 1 minute to read in a 6GB .csv file on a 64GB machine. Variable names available at: https://www.hcup-us.ahrq.gov/db/nation/neds/nedssasloadprog.jsp#2006v3. Warnings about bumping the ICD9's and Ecodes from numeric to character.

```r
names(neds.2008.core) <- c("AGE", "AMONTH", "AWEEKEND", "CHRON1",
    "CHRON2", "CHRON3", "CHRON4", "CHRON5", "CHRON6", "CHRON7",
    "CHRON8", "CHRON9", "CHRON10", "CHRON11", "CHRON12", "CHRON13",
    "CHRON14", "CHRON15", "DIED_VISIT", "DISCWT", "DISP_ED",
    "DQTR", "DX1", "DX2", "DX3", "DX4", "DX5", "DX6", "DX7",
    "DX8", "DX9", "DX10", "DX11", "DX12", "DX13", "DX14", "DX15",
    "DXCCS1", "DXCCS2", "DXCCS3", "DXCCS4", "DXCCS5", "DXCCS6",
    "DXCCS7", "DXCCS8", "DXCCS9", "DXCCS10", "DXCCS11", "DXCCS12",
    "DXCCS13", "DXCCS14", "DXCCS15", "ECODE1", "ECODE2", "ECODE3",
    "ECODE4", "EDEVENT", "E_CCS1", "E_CCS2", "E_CCS3", "E_CCS4",
    "FEMALE", "HCUPFILE", "HOSP_ED", "HOSP_REGION", "INTENT_SELF_HARM",
    "KEY_ED", "NDX", "NECODE", "NEDS_STRATUM", "PAY1", "PAY2",
    "PL_NCHS2006", "TOTCHG_ED", "YEAR", "ZIPINC_QRTL")
saveRDS(neds.2008.core, "~/2008/NEDS_2008_Core.rds")
nis.2008.core <- readRDS("~/2008/NEDS_2008_Core.rds")


neds.2009.core <- fread("~/2009/NEDS_2009_Core.csv")
names(neds.2009.core) <- c("AGE", "AMONTH", "AWEEKEND", "CHRON1",
    "CHRON2", "CHRON3", "CHRON4", "CHRON5", "CHRON6", "CHRON7",
    "CHRON8", "CHRON9", "CHRON10", "CHRON11", "CHRON12", "CHRON13",
    "CHRON14", "CHRON15", "DIED_VISIT", "DISCWT", "DISP_ED",
    "DQTR", "DX1", "DX2", "DX3", "DX4", "DX5", "DX6", "DX7",
    "DX8", "DX9", "DX10", "DX11", "DX12", "DX13", "DX14", "DX15",
    "DXCCS1", "DXCCS2", "DXCCS3", "DXCCS4", "DXCCS5", "DXCCS6",
    "DXCCS7", "DXCCS8", "DXCCS9", "DXCCS10", "DXCCS11", "DXCCS12",
    "DXCCS13", "DXCCS14", "DXCCS15", "ECODE1", "ECODE2", "ECODE3",
    "ECODE4", "EDEVENT", "E_CCS1", "E_CCS2", "E_CCS3", "E_CCS4",
    "FEMALE", "HCUPFILE", "HOSP_ED", "HOSP_REGION", "INJURY",
    "INJURY_CUT", "INJURY_DROWN", "INJURY_FALL", "INJURY_FIRE",
    "INJURY_FIREARM", "INJURY_MACHINERY", "INJURY_MVT", "INJURY_NATURE",
    "INJURY_POISON", "INJURY_SEVERITY", "INJURY_STRUCK", "INJURY_SUFFOCATION",
    "INTENT_ASSAULT", "INTENT_SELF_HARM", "INTENT_UNINTENTIONA",
    "KEY_ED", "MULTINJURY", "NDX", "NECODE", "NEDS_STRATUM",
    "PAY1", "PAY2", "PL_NCHS2006", "TOTCHG_ED", "YEAR", "ZIPINC_QRTL")
saveRDS(neds.2009.core, "~/2009/NEDS_2009_Core.rds")
nis.2009.core <- readRDS("~/2009/NEDS_2009_Core.rds")

neds.2010.core <- fread("~/NEDS_2010_Core.csv")
names(neds.2010.core) <- c("AGE", "AMONTH", "AWEEKEND", "CHRON1",
    "CHRON2", "CHRON3", "CHRON4", "CHRON5", "CHRON6", "CHRON7",
    "CHRON8", "CHRON9", "CHRON10", "CHRON11", "CHRON12", "CHRON13",
    "CHRON14", "CHRON15", "DIED_VISIT", "DISCWT", "DISP_ED",
    "DQTR", "DX1", "DX2", "DX3", "DX4", "DX5", "DX6", "DX7",
    "DX8", "DX9", "DX10", "DX11", "DX12", "DX13", "DX14", "DX15",
    "DXCCS1", "DXCCS2", "DXCCS3", "DXCCS4", "DXCCS5", "DXCCS6",
    "DXCCS7", "DXCCS8", "DXCCS9", "DXCCS10", "DXCCS11", "DXCCS12",
```

```
    "DXCCS13", "DXCCS14", "DXCCS15", "ECODE1", "ECODE2", "ECODE3",
    "ECODE4", "EDEVENT", "E_CCS1", "E_CCS2", "E_CCS3", "E_CCS4",
    "FEMALE", "HCUPFILE", "HOSP_ED", "HOSP_REGION", "INJURY",
    "INJURY_CUT", "INJURY_DROWN", "INJURY_FALL", "INJURY_FIRE",
    "INJURY_FIREARM", "INJURY_MACHINERY", "INJURY_MVT", "INJURY_NATURE",
    "INJURY_POISON", "INJURY_SEVERITY", "INJURY_STRUCK", "INJURY_SUFFOCATION",
    "INTENT_ASSAULT", "INTENT_SELF_HARM", "INTENT_UNINTENTIONA",
    "KEY_ED", "MULTINJURY", "NDX", "NECODE", "NEDS_STRATUM",
    "PAY1", "PAY2", "PL_NCHS2006", "TOTCHG_ED", "YEAR", "ZIPINC_QRTL")
saveRDS(neds.2010.core, "~/2010/NEDS_2010_Core.rds")
nis.2010.core <- readRDS("~/2010/NEDS_2010_Core.rds")


neds.2011.core <- fread("~/NEDS_2011_Core.csv")
names(neds.2011.core) <- c("AGE", "AMONTH", "AWEEKEND", "CHRON1",
    "CHRON2", "CHRON3", "CHRON4", "CHRON5", "CHRON6", "CHRON7",
    "CHRON8", "CHRON9", "CHRON10", "CHRON11", "CHRON12", "CHRON13",
    "CHRON14", "CHRON15", "DIED_VISIT", "DISCWT", "DISP_ED",
    "DQTR", "DX1", "DX2", "DX3", "DX4", "DX5", "DX6", "DX7",
    "DX8", "DX9", "DX10", "DX11", "DX12", "DX13", "DX14", "DX15",
    "DXCCS1", "DXCCS2", "DXCCS3", "DXCCS4", "DXCCS5", "DXCCS6",
    "DXCCS7", "DXCCS8", "DXCCS9", "DXCCS10", "DXCCS11", "DXCCS12",
    "DXCCS13", "DXCCS14", "DXCCS15", "ECODE1", "ECODE2", "ECODE3",
    "ECODE4", "EDEVENT", "E_CCS1", "E_CCS2", "E_CCS3", "E_CCS4",
    "FEMALE", "HCUPFILE", "HOSP_ED", "INJURY", "INJURY_CUT",
    "INJURY_DROWN", "INJURY_FALL", "INJURY_FIRE", "INJURY_FIREARM",
    "INJURY_MACHINERY", "INJURY_MVT", "INJURY_NATURE", "INJURY_POISON",
    "INJURY_SEVERITY", "INJURY_STRUCK", "INJURY_SUFFOCATION",
    "INTENT_ASSAULT", "INTENT_SELF_HARM", "INTENT_UNINTENTION",
    "KEY_ED", "MULTINJURY", "NDX", "NECODE", "NEDS_STRATUM",
    "PAY1", "PAY2", "PL_NCHS2006", "TOTCHG_ED", "YEAR", "ZIPINC_QRTL")
saveRDS(neds.2011.core, "~/2011/NEDS_2011_Core.rds")
nis.2011.core <- readRDS("~/2011/NEDS_2011_Core.rds")


neds.2012.core <- fread("~/NEDS_2012_Core.csv")
names(neds.2012.core) <- c("AGE", "AMONTH", "AWEEKEND", "CHRON1",
    "CHRON2", "CHRON3", "CHRON4", "CHRON5", "CHRON6", "CHRON7",
    "CHRON8", "CHRON9", "CHRON10", "CHRON11", "CHRON12", "CHRON13",
    "CHRON14", "CHRON15", "DIED_VISIT", "DISCWT", "DISP_ED",
    "DQTR", "DX1", "DX2", "DX3", "DX4", "DX5", "DX6", "DX7",
    "DX8", "DX9", "DX10", "DX11", "DX12", "DX13", "DX14", "DX15",
    "DXCCS1", "DXCCS2", "DXCCS3", "DXCCS4", "DXCCS5", "DXCCS6",
    "DXCCS7", "DXCCS8", "DXCCS9", "DXCCS10", "DXCCS11", "DXCCS12",
    "DXCCS13", "DXCCS14", "DXCCS15", "ECODE1", "ECODE2", "ECODE3",
    "ECODE4", "EDEVENT", "E_CCS1", "E_CCS2", "E_CCS3", "E_CCS4",
    "FEMALE", "HCUPFILE", "HOSP_ED", "INJURY", "INJURY_CUT",
```

```
    "INJURY_DROWN", "INJURY_FALL", "INJURY_FIRE", "INJURY_FIREARM",
    "INJURY_MACHINERY", "INJURY_MVT", "INJURY_NATURE", "INJURY_POISON",
    "INJURY_SEVERITY", "INJURY_STRUCK", "INJURY_SUFFOCATION",
    "INTENT_ASSAULT", "INTENT_SELF_HARM", "INTENT_UNINTENTIONA",
    "KEY_ED", "MULTINJURY", "NDX", "NECODE", "NEDS_STRATUM",
    "PAY1", "PAY2", "PL_NCHS2006", "TOTCHG_ED", "YEAR", "ZIPINC_QRTL")
saveRDS(neds.2012.core, "~/2012/NEDS_2012_Core.rds")
nis.2012.core <- readRDS("~/2012/NEDS_2012_Core.rds")
```

## 4.2 Select Out Injuries

Use the first-listed diagnosis to identify and restrict to injuries. [5]

```
neds.2006.core <- readRDS("~/2006/NEDS_2006_Core.rds")
neds.2006.inj <- neds.2006.core[neds.2006.core$DX1 %in% 800:996 |
    neds.2006.core$DX1 %in% 8000:9960 | neds.2006.core$DX1 %in%
    80000:99600, ]
saveRDS(neds.2006.inj, "~/neds_2006_inj.rds")
nrow(neds.2006.inj)  # 6,109,718
summary(neds.2006.inj$AGE)  # 33.82
rm(neds.2006.core, neds.2006.inj)

neds.2007.core <- readRDS("~/2007/NEDS_2007_Core.rds")
neds.2007.inj <- neds.2007.core[neds.2007.core$DX1 %in% 800:996 |
    neds.2007.core$DX1 %in% 8000:9960 | neds.2007.core$DX1 %in%
    80000:99600, ]
saveRDS(neds.2007.inj, "~/neds_2007_inj.rds")
nrow(neds.2007.inj)  #  6,143,514
summary(neds.2007.inj$AGE)  # 34.03
rm(neds.2007.core, neds.2007.inj)

neds.2008.core <- readRDS("~/2008/NEDS_2008_Core.rds")
neds.2008.inj <- neds.2008.core[neds.2008.core$DX1 %in% 800:996 |
    neds.2008.core$DX1 %in% 8000:9960 | neds.2008.core$DX1 %in%
    80000:99600, ]
saveRDS(neds.2008.inj, "~/neds_2008_inj.rds")
nrow(neds.2008.inj)  # 6,360,564
summary(neds.2008.inj$AGE)  # 34.51
rm(neds.2008.core, neds.2008.inj)


neds.2009.core <- readRDS("~/2009/NEDS_2009_Core.rds")
neds.2009.inj <- neds.2009.core[neds.2009.core$DX1 %in% 800:996 |
```

[5] See "HCUP Methods Series: Special Study on the Meaning of the First Listed Diagnosis". For outpatient and ED services, "UB-04 coding manual ... guides hospitals to use the "first-listed diagnosis" in lieu of "principal diagnosis"

```r
    neds.2009.core$DX1 %in% 8000:9960 | neds.2009.core$DX1 %in%
    80000:99600, ]
saveRDS(neds.2009.inj, "~/neds_2009_inj.rds")
nrow(neds.2009.inj)  # 6,087,563
summary(neds.2009.inj$AGE)  # 34.7
rm(neds.2009.core, neds.2009.inj)


neds.2010.core <- readRDS("~/2010/NEDS_2010_Core.rds")
neds.2010.inj <- neds.2010.core[neds.2010.core$DX1 %in% 800:996 |
    neds.2010.core$DX1 %in% 8000:9960 | neds.2010.core$DX1 %in%
    80000:99600, ]
saveRDS(neds.2010.inj, "~/neds_2010_inj.rds")
nrow(neds.2010.inj)  # 6135873
summary(neds.2010.inj$AGE)  # 35.15
rm(neds.2010.core, neds.2010.inj)


neds.2011.core <- readRDS("~/2011/NEDS_2011_Core.rds")
neds.2011.inj <- neds.2011.core[neds.2011.core$DX1 %in% 800:996 |
    neds.2011.core$DX1 %in% 8000:9960 | neds.2011.core$DX1 %in%
    80000:99600, ]
saveRDS(neds.2011.inj, "~/neds_2011_inj.rds")
nrow(neds.2011.inj)  # 6022850
summary(neds.2011.inj$AGE)  # 35.2
rm(neds.2011.core, neds.2011.inj)



neds.2012.core <- readRDS("~/2012/NEDS_2012_Core.rds")
neds.2012.inj <- neds.2012.core[neds.2012.core$DX1 %in% 800:996 |
    neds.2012.core$DX1 %in% 8000:9960 | neds.2012.core$DX1 %in%
    80000:99600, ]
saveRDS(neds.2012.inj, "~/neds_2012_inj.rds")
nrow(neds.2012.inj)  # 6339263
summary(neds.2012.inj$AGE)  # 35.2
rm(neds.2012.core, neds.2012.inj)
```

Bind the individual year files into a single file using data.table::rbindlist() with option to match columns by name (rather than position, which is default); fill in non-matching columns with NAs. [6]. This should result in a **full file of 43,199,345 observations and 93 variables**.

```r
ls()

neds.2006.inj <- readRDS("~/neds_2006_inj.rds")
```

[6] This is the same behavior as plyr::rbind.fill()

```
neds.2007.inj <- readRDS("~/neds_2007_inj.rds")
neds.2008.inj <- readRDS("~/neds_2008_inj.rds")
neds.2009.inj <- readRDS("~/neds_2009_inj.rds")
neds.2010.inj <- readRDS("~/neds_2010_inj.rds")
neds.2011.inj <- readRDS("~/neds_2011_inj.rds")
neds.2012.inj <- readRDS("~/neds_2012_inj.rds")

neds.98_12.inj <- rbindlist(list(neds.2006.inj, neds.2007.inj,
    neds.2008.inj, neds.2009.inj, neds.2010.inj, neds.2011.inj,
    neds.2012.inj), use.names = TRUE, fill = TRUE)

str(neds.98_12.inj)  #  43,199,345 obs 93 variables

saveRDS(neds.98_12.inj, "~/neds.06_12.inj.rds")

rm(neds.2006.inj, neds.2007.inj, neds.2008.inj, neds.2009.inj,
    neds.2010.inj, neds.2011.inj, neds.2012.inj)
```

Use the R "icd9" package to apply descriptors to the diagnostic codes. Clean and restrict to acute traumatic injuries.

```
inj <- readRDS("~/neds.06_12.inj.rds")

icd.tab1 <- as.data.frame(table(inj$DX1))

nrow(icd.tab1)  #2367

icd9Explain(icd.tab1$Var1, brief = T)

# use icd9::icd9Explain() to add dx descriptions (note: loop
# because apply would not accept icd9Explain as a fx...)
for (i in 1:nrow(icd.tab1)) {
    icd.tab1$descriptor[i] <- icd9Explain(icd.tab1$Var1[i], brief = T)
}
head(icd.tab1)
icd.tab1
```

Restrict to acute injuries by removing "late effect" primary diagnoses (ICD 9050-9099), insect bites, poisonings, anaphylaxis and some additional miscellaneous diagnoses (malignant hyperthermia, systemic inflammatory response syndrome, malfunctioning cardiac devices). Add age group variable.

```
lateEffects<-as.character(c(9050:9099))
insectBites <- as.character(c(9104, 9105, 9114, 9115, 9124, 9125, 9134, 9135, 9144, 9145,
9154, 9155, 9164, 9165, 9174, 9175, 9194, 9195))
poisoning<-as.character(c(9600:9649, 96500:96502, 96509, 9651, 9654, 9655, 96561, 96569,
9657:9690, 96900:96909, 96970:96973, 9691:9697, 9670:9679, 96979, 9698:9809, 97081,
97089, 981, 9820:9859, 986, 9870:9897, 98981:98989, 9899, 990, 9910:9952,
```

```
99520:99529, 9953, 9954))
anaphylaxis<-as.character(c(99560:99569,9957))
misc<-as.character(c(99586:99600))

non.trauma<-c(lateEffects, insectBites, poisoning, anaphylaxis, misc)

inj2<-inj[! inj$DX1 %in% non.trauma,]

create file so ecode counts
icd.tab2<-as.data.frame(table(inj2$DX1))

nrow(icd.tab2) #1994

for(i in 1:nrow(icd.tab2)){
icd.tab2$descriptor[i]<-icd9Explain(icd.tab2$Var1[i], brief=T)
}

head(icd.tab2)

saveRDS(icd.tab2, "/Users/charlie/Box Sync/hcup/NEDS/nedsData/neds.06_12.icd9Counts.rds")

# create age groups
inj2$ageGrp[inj2$AGE<18]<-"1.LT18"
inj2$ageGrp[inj2$AGE>=18 & inj2$AGE<45]<-"2.GT17.LT45"
inj2$ageGrp[inj2$AGE>=45 & inj2$AGE<65]<-"3.GT44.LT65"
inj2$ageGrp[inj2$AGE>=65 & inj2$AGE<85]<-"4.GT64.LT85"
inj2$ageGrp[inj2$AGE>84]<-"5.GT84"


# survey doesn't seem to like non-numeric grouping variablews sometimes
inj2$ageGrp.num[inj2$ageGrp== "1.LT18" ]<-1
inj2$ageGrp.num[inj2$ageGrp== "2.GT17.LT45" ]<-2
inj2$ageGrp.num[inj2$ageGrp== "3.GT44.LT65" ]<-3
inj2$ageGrp.num[inj2$ageGrp== "4.GT64.LT85" ]<-4
inj2$ageGrp.num[inj2$ageGrp== "5.GT84" ]<-5

saveRDS(inj2, "~/neds.06_12.inj.rds")
```

## 4.3 Add Barell Matrix Diagnostic Codes

Map ICD9 diagnostic codes to the Barell Matrix.

```
inj.sub <- inj

# create 3 input variables (NB: change from CDC SAS Code,
# using DX1.3, 1.4, 1.4 because DX13, 14,15 already exist in
```

```
# the data)
inj.sub$DX1.3 <- substr(inj.sub$DX1, 1, 3)  # [3 digit code for first-listed ICD-9 CM ]
inj.sub$DX1.4 <- ifelse(nchar(inj.sub$DX1) == 4, inj.sub$DX1,
    NA)  # [4 digit code for first-listed diagnosis]
inj.sub$DX1.5 <- ifelse(nchar(inj.sub$DX1) == 5, inj.sub$DX1,
    NA)  # [5 digit code for first-listed diagnosis]
inj.sub$D5 <- ifelse(nchar(inj.sub$DX1) == 5, substr(inj.sub$DX1,
    5, 5), NA)  # [5th digit of code]


# functions to return values in a range (replicate the SAS
# code)
"%betw1%" <- function(x, ends) ends[1] <= x & x <= ends[2]
"%betw2%" <- function(x, ends) x >= ends[1] & x <= ends[2]  # for the GE LE syntax

# ISRCODE

inj.sub$ISRCODE[inj.sub$DX1.3 %betw1% c("800", "829")] <- 1
inj.sub$ISRCODE[inj.sub$DX1.3 %betw2% c("830", "839")] <- 2
inj.sub$ISRCODE[inj.sub$DX1.3 %betw2% c("840", "848")] <- 3
inj.sub$ISRCODE[inj.sub$DX1.3 %betw1% c("860", "869") | inj.sub$DX1.3 %betw1%
    c("850", "854") | inj.sub$DX1.3 == "952" | inj.sub$DX1.5 ==
    "99555"] <- 4
inj.sub$ISRCODE[inj.sub$DX1.3 %betw1% c("870", "884") | inj.sub$DX1.3 %betw1%
    c("890", "894")] <- 5
inj.sub$ISRCODE[inj.sub$DX1.3 %betw1% c("885", "887") | inj.sub$DX1.3 %betw1%
    c("895", "897")] <- 6
inj.sub$ISRCODE[inj.sub$DX1.3 %betw2% c("900", "904")] <- 7
inj.sub$ISRCODE[inj.sub$DX1.3 %betw2% c("910", "924")] <- 8
inj.sub$ISRCODE[inj.sub$DX1.3 %betw2% c("925", "929")] <- 9
inj.sub$ISRCODE[inj.sub$DX1.3 %betw2% c("940", "949")] <- 10
inj.sub$ISRCODE[inj.sub$DX1.3 %betw2% c("950", "951") | inj.sub$DX1.3 %betw1%
    c("953", "957")] <- 11
inj.sub$ISRCODE[inj.sub$DX1.3 == "959"] <- 12
inj.sub$ISRCODE[inj.sub$DX1.3 %betw1% c("930", "939") | inj.sub$DX1.3 %betw1%
    c("960", "994") | inj.sub$DX1.3 %betw1% c("905", "908") |
    inj.sub$DX1.4 %betw1% c("9090", "9092") | inj.sub$DX1.3 ==
    "958" | inj.sub$DX1.5 %betw1% c("99550", "99554") | inj.sub$DX1.5 ==
    "99559" | inj.sub$DX14 == "9094" | inj.sub$DX1.4 == "9099" |
    inj.sub$DX1.5 %betw1% c("99580", "99585")] <- 13

table(inj.sub$ISRCODE)


# ISRSITE

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8001", "8004") | inj.sub$DX1.4 %betw1%
```

```r
    c("8006", "8009") | inj.sub$DX1.5 %betw1% c("80003", "80005") |
    inj.sub$DX1.5 %betw1% c("80053", "80055") | inj.sub$DX1.4 %betw1%
    c("8011", "8014") | inj.sub$DX1.4 %betw1% c("8016", "8019") |
    inj.sub$DX1.5 %betw1% c("80103", "80105") | inj.sub$DX1.5 %betw1%
    c("80153", "80155") | inj.sub$DX1.4 %betw1% c("8031", "8034") |
    inj.sub$DX1.4 %betw1% c("8036", "8039") | inj.sub$DX1.5 %betw1%
    c("80303", "80305") | inj.sub$DX1.5 %betw1% c("80353", "80355") |
    inj.sub$DX1.4 %betw1% c("8041", "8044") | inj.sub$DX1.4 %betw1%
    c("8046", "8049") | inj.sub$DX1.5 %betw1% c("80403", "80405") |
    inj.sub$DX1.5 %betw1% c("80453", "80455") | inj.sub$DX1.4 %betw1%
    c("8502", "8504") | inj.sub$DX1.3 %betw1% c("851", "854") |
    inj.sub$DX1.4 %betw1% c("9501", "9503") | inj.sub$DX1.5 ==
    "99555"] <- 1

inj.sub$ISRSITE[inj.sub$DX1.5 == "80000" | inj.sub$DX1.5 == "80002" |
    inj.sub$DX1.5 == "80006" | inj.sub$DX1.5 == "80009" | inj.sub$DX1.5 ==
    "80100" | inj.sub$DX1.5 == "80102" | inj.sub$DX1.5 == "80106" |
    inj.sub$DX1.5 == "80109" | inj.sub$DX1.5 == "80300" | inj.sub$DX1.5 ==
    "80302" | inj.sub$DX1.5 == "80306" | inj.sub$DX1.5 == "80309" |
    inj.sub$DX1.5 == "80400" | inj.sub$DX1.5 == "80402" | inj.sub$DX1.5 ==
    "80406" | inj.sub$DX1.5 == "80409" | inj.sub$DX1.5 == "80050" |
    inj.sub$DX1.5 == "80052" | inj.sub$DX1.5 == "80056" | inj.sub$DX1.5 ==
    "80059" | inj.sub$DX1.5 == "80150" | inj.sub$DX1.5 == "80152" |
    inj.sub$DX1.5 == "80156" | inj.sub$DX1.5 == "80159" | inj.sub$DX1.5 ==
    "80350" | inj.sub$DX1.5 == "80352" | inj.sub$DX1.5 == "80356" |
    inj.sub$DX1.5 == "80359" | inj.sub$DX1.5 == "80450" | inj.sub$DX1.5 ==
    "80452" | inj.sub$DX1.5 == "80456" | inj.sub$DX1.5 == "80459" |
    inj.sub$DX1.4 == "8500" | inj.sub$DX1.4 == "8501" | inj.sub$DX1.4 ==
    "8505" | inj.sub$DX1.4 == "8509"] <- 2

inj.sub$ISRSITE[inj.sub$DX1.5 == "80001" | inj.sub$DX1.5 == "80051" |
    inj.sub$DX1.5 == "80101" | inj.sub$DX1.5 == "80151" | inj.sub$DX1.5 ==
    "80301" | inj.sub$DX1.5 == "80351" | inj.sub$DX1.5 == "80401" |
    inj.sub$DX1.5 == "80451"] <- 3

inj.sub$ISRSITE[inj.sub$DX1.3 == "951" | inj.sub$DX1.4 == "8730" |
    inj.sub$DX1.4 == "8731" | inj.sub$DX1.4 == "8738" | inj.sub$DX1.4 ==
    "8739" | (inj.sub$DX1.3 == "941" & inj.sub$DX5 == "6") |
    inj.sub$DX1.5 == "95901"] <- 4

inj.sub$ISRSITE[inj.sub$DX1.3 == "802" | inj.sub$DX1.3 == "830" |
    inj.sub$DX1.4 == "8480" | inj.sub$DX1.4 == "8481" | inj.sub$DX1.3 ==
    "872" | inj.sub$DX1.4 %betw1% c("8732", "8737") | (inj.sub$DX1.3 ==
    "941" & inj.sub$DX5 == "1") | (inj.sub$DX1.3 == "941" & inj.sub$DX5 %betw1%
    c("3", "5")) | (inj.sub$DX1.3 == "941" & inj.sub$DX5 == "7")] <- 5

inj.sub$ISRSITE[inj.sub$DX1.4 == "9500" | inj.sub$DX1.4 == "9509" |
```

```
    inj.sub$DX1.3 %betw1% c("870", "871") | inj.sub$DX1.3 ==
    "921" | inj.sub$DX1.3 == "918" | inj.sub$DX1.3 == "940" |
    (inj.sub$DX1.3 == "941" & inj.sub$DX5 == "2")] <- 6

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8075", "8076") | inj.sub$DX1.4 ==
    "8482" | inj.sub$DX1.4 == "9252" | inj.sub$DX1.4 == "9530" |
    inj.sub$DX1.4 == "9540" | inj.sub$DX1.3 == "874" | (inj.sub$DX1.3 ==
    "941" & inj.sub$DX5 == "8")] <- 7

inj.sub$ISRSITE[inj.sub$DX1.4 == "9251" | inj.sub$DX1.3 == "900" |
    inj.sub$DX1.4 == "9570" | inj.sub$DX1.3 == "910" | inj.sub$DX1.3 ==
    "920" | inj.sub$DX1.4 == "9470" | inj.sub$DX1.5 == "95909" |
    (inj.sub$DX1.3 == "941" & inj.sub$DX5 == "0") | inj.sub$DX5 ==
    "9"] <- 8

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8060", "8061") | inj.sub$DX1.4 ==
    "9520"] <- 9

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8062", "8063") | inj.sub$DX1.4 ==
    "9521"] <- 10

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8064", "8065") | inj.sub$DX1.4 ==
    "9522"] <- 11

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8066", "8067") | inj.sub$DX1.4 %betw1%
    c("9523", "9524")] <- 12

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8068", "8069") | inj.sub$DX1.4 %betw1%
    c("9528", "9529")] <- 13

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8050", "8051") | inj.sub$DX1.4 %betw1%
    c("8390", "8391") | inj.sub$DX1.4 == "8470"] <- 14

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8052", "8053") | "83921" ==
    inj.sub$DX1.5 | "83931" == inj.sub$DX1.5 | inj.sub$DX1.4 ==
    "8471"] <- 15

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8054", "8055") | "83920" ==
    inj.sub$DX1.5 | "83930" == inj.sub$DX1.5 | inj.sub$DX1.4 ==
    "8472"] <- 16

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8056", "8057") | "83941" ==
    inj.sub$DX1.5 | "83942" == inj.sub$DX1.5 | inj.sub$DX1.5 %betw1%
    c("83951", "83952") | inj.sub$DX1.4 %betw1% c("8473,8474")] <- 17

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8058", "8059") | "83940" ==
    inj.sub$DX1.5 | "83949" == inj.sub$DX1.5 | "83950" == inj.sub$DX1.5 |
```

```r
    inj.sub$DX1.5 == "83959"] <- 18

inj.sub$ISRSITE[inj.sub$DX1.4 %betw1% c("8070", "8074") | inj.sub$DX1.5 ==
    "83961" | inj.sub$DX1.5 == "83971" | inj.sub$DX1.4 %betw1%
    c("8483", "8484") | inj.sub$DX1.5 == "92619" | inj.sub$DX1.3 %betw1%
    c("860", "862") | inj.sub$DX1.3 == "901" | inj.sub$DX1.4 ==
    "9531" | inj.sub$DX1.3 == "875" | inj.sub$DX1.4 == "8790" |
    inj.sub$DX1.4 == "8791" | inj.sub$DX1.4 == "9220" | inj.sub$DX1.4 ==
    "9221" | inj.sub$DX1.5 == "92233" | (inj.sub$DX1.3 == "942" &
    inj.sub$DX5 == "1") | inj.sub$DX5 == "2"] <- 19

inj.sub$ISRSITE[inj.sub$DX1.3 %betw1% c("863", "866") | inj.sub$DX1.3 ==
    "868" | inj.sub$DX1.4 %betw1% c("9020", "9024") | inj.sub$DX1.4 ==
    "9532" | inj.sub$DX1.4 == "9535" | inj.sub$DX1.4 %betw1%
    c("8792", "8795") | inj.sub$DX1.4 == "9222" | (inj.sub$DX1.3 ==
    "942" & inj.sub$DX5 == "3") | inj.sub$DX1.4 == "9473"] <- 20

inj.sub$ISRSITE[inj.sub$DX1.3 == "808" | inj.sub$DX1.5 == "83969" |
    inj.sub$DX1.5 == "83979" | inj.sub$DX1.3 == "846" | inj.sub$DX1.4 ==
    "8485" | inj.sub$DX1.4 == "9260" | inj.sub$DX1.5 == "92612" |
    inj.sub$DX1.3 == "867" | inj.sub$DX1.4 == "9025" | inj.sub$DX1.5 %betw1%
    c("90281", "90282") | inj.sub$DX1.4 == "9533" | inj.sub$DX1.3 %betw1%
    c("877", "878") | inj.sub$DX1.4 == "9224" | (inj.sub$DX1.3 ==
    "942" & inj.sub$DX5 == "5") | inj.sub$DX1.4 == "9474"] <- 21

inj.sub$ISRSITE[inj.sub$DX1.3 == "809" | inj.sub$DX1.4 %betw1%
    c("9268", "9269") | inj.sub$DX1.4 == "9541" | inj.sub$DX1.4 %betw1%
    c("9548,9549") | inj.sub$DX1.4 %betw1% c("8796", "8797") |
    inj.sub$DX1.4 %betw1% c("9228,9229") | inj.sub$DX1.3 == "911" |
    (inj.sub$DX1.3 == "942" & inj.sub$DX5 == "9") | inj.sub$DX1.4 ==
    "9591"] <- 22

inj.sub$ISRSITE[inj.sub$DX1.4 == "8479" | inj.sub$DX1.5 == "92611" |
    inj.sub$DX1.3 == "876" | inj.sub$DX1.5 == "92232" | inj.sub$DX1.5 ==
    "92231" | (inj.sub$DX1.3 == "942" & inj.sub$DX5 == "4")] <- 23

inj.sub$ISRSITE[inj.sub$DX1.3 %betw1% c("810", "812") | inj.sub$DX1.3 ==
    "831" | inj.sub$DX1.3 == "840" | inj.sub$DX1.3 == "880" |
    inj.sub$DX1.4 %betw1% c("8872", "8873") | (inj.sub$DX1.3 ==
    "943" & inj.sub$DX5 %betw1% c("3", "6")) | inj.sub$DX1.3 ==
    "912" | inj.sub$DX1.4 == "9230" | inj.sub$DX1.4 == "9270" |
    inj.sub$DX1.4 == "9592"] <- 24

inj.sub$ISRSITE[inj.sub$DX1.3 == "813" | inj.sub$DX1.3 == "832" |
    inj.sub$DX1.3 == "841" | (inj.sub$DX1.3 == "881" & inj.sub$DX5 %betw1%
    c("0", "1")) | inj.sub$DX1.4 %betw1% c("8870", "8871") |
    inj.sub$DX1.4 == "9231" | inj.sub$DX1.4 == "9271" | (inj.sub$DX1.3 ==
```

```
    "943" & inj.sub$DX5 %betw1% c("1", "2"))] <- 25

inj.sub$ISRSITE[inj.sub$DX1.3 %betw1% c("814", "817") | inj.sub$DX1.3 %betw1%
    c("833", "834") | inj.sub$DX1.3 == "842" | (inj.sub$DX1.3 ==
    "881" & inj.sub$DX5 == "2") | inj.sub$DX1.3 %betw1% c("882",
    "883") | inj.sub$DX1.3 %betw1% c("885", "886") | inj.sub$DX1.3 %betw1%
    c("914", "915") | inj.sub$DX1.4 %betw1% c("9232", "9233") |
    inj.sub$DX1.4 %betw1% c("9272", "9273") | inj.sub$DX1.3 ==
    "944" | inj.sub$DX1.4 %betw1% c("9594", "9595")] <- 26

inj.sub$ISRSITE[inj.sub$DX1.3 == "818" | inj.sub$DX1.3 == "884" |
    inj.sub$DX1.4 %betw1% c("8874", "8877") | inj.sub$DX1.3 ==
    "903" | inj.sub$DX1.3 == "913" | inj.sub$DX1.4 == "9593" |
    inj.sub$DX1.4 %betw1% c("9238", "9239") | inj.sub$DX1.4 %betw1%
    c("9278", "9279") | inj.sub$DX1.4 == "9534" | inj.sub$DX1.3 ==
    "955" | (inj.sub$DX1.3 == "943" & inj.sub$DX5 == "0") | inj.sub$DX5 ==
    "9"] <- 27

inj.sub$ISRSITE[inj.sub$DX1.3 == "820" | inj.sub$DX1.3 == "835" |
    inj.sub$DX1.3 == "843" | inj.sub$DX1.5 == "92401" | inj.sub$DX1.5 ==
    "92801"] <- 28

inj.sub$ISRSITE[inj.sub$DX1.3 == "821" | inj.sub$DX1.4 %betw1%
    c("8972", "8973") | inj.sub$DX1.5 == "92400" | inj.sub$DX1.5 ==
    "92800" | (inj.sub$DX1.3 == "945" & inj.sub$DX5 == "6")] <- 29

inj.sub$ISRSITE[inj.sub$DX1.3 == "822" | inj.sub$DX1.3 == "836" |
    inj.sub$DX1.4 %betw1% c("8440", "8443") | inj.sub$DX1.5 ==
    "92411" | inj.sub$DX1.5 == "92811" | (inj.sub$DX1.3 == "945" &
    inj.sub$DX5 == "5")] <- 30

inj.sub$ISRSITE[inj.sub$DX1.3 %betw1% c("823", "824") | inj.sub$DX1.4 %betw1%
    c("8970", "8971") | inj.sub$DX1.3 == "837" | inj.sub$DX1.4 ==
    "8450" | inj.sub$DX1.5 == "92410" | inj.sub$DX1.5 == "92421" |
    inj.sub$DX1.5 == "92810" | inj.sub$DX1.5 == "92821" | (inj.sub$DX1.3 ==
    "945" & inj.sub$DX5 %betw1% c("3", "4"))] <- 31

inj.sub$ISRSITE[inj.sub$DX1.3 %betw1% c("825", "826") | inj.sub$DX1.3 ==
    "838" | inj.sub$DX1.4 == "8451" | inj.sub$DX1.3 %betw1% c("892",
    "893") | inj.sub$DX1.3 %betw1% c("895", "896") | inj.sub$DX1.3 ==
    "917" | inj.sub$DX1.5 == "92420" | inj.sub$DX1.4 == "9243" |
    inj.sub$DX1.5 == "92820" | inj.sub$DX1.4 == "9283" | (inj.sub$DX1.3 ==
    "945" & inj.sub$DX5 %betw1% c("1", "2"))] <- 32

inj.sub$ISRSITE[inj.sub$DX1.3 == "827" | inj.sub$DX1.4 %betw1%
    c("8448", "8449") | inj.sub$DX1.3 %betw1% c("890", "891") |
    inj.sub$DX1.3 == "894" | inj.sub$DX1.4 %betw1% c("8974",
```

```r
    "8977") | inj.sub$DX1.4 %betw1% c("9040", "9048") | inj.sub$DX1.3 ==
    "916" | inj.sub$DX1.4 %betw1% c("9244", "9245") | inj.sub$DX1.4 ==
    "9288" | inj.sub$DX1.4 == "9289" | inj.sub$DX1.4 %betw1%
    c("9596", "9597") | (inj.sub$DX1.3 == "945" & inj.sub$DX5 ==
    "0") | inj.sub$DX5 == "9"] <- 33

inj.sub$ISRSITE[inj.sub$DX1.3 == "828" | inj.sub$DX1.3 == "819" |
    inj.sub$DX1.5 == "90287" | inj.sub$DX1.5 == "90289" | inj.sub$DX1.4 ==
    "9538" | inj.sub$DX1.4 %betw1% c("9471", "9472") | inj.sub$DX1.3 ==
    "956"] <- 34

inj.sub$ISRSITE[inj.sub$DX1.3 == "829" | inj.sub$DX1.4 %betw1%
    c("8398", "8399") | inj.sub$DX1.4 %betw1% c("8488", "8489") |
    inj.sub$DX1.3 == "869" | inj.sub$DX1.4 %betw1% c("8798",
    "8799") | inj.sub$DX1.4 == "9029" | inj.sub$DX1.4 == "9049" |
    inj.sub$DX1.3 == "919" | inj.sub$DX1.4 %betw1% c("9248",
    "9249") | inj.sub$DX1.3 == "929" | inj.sub$DX1.3 == "946" |
    inj.sub$DX1.4 %betw1% c("9478", "9479") | inj.sub$DX1.3 %betw1%
    c("948", "949") | inj.sub$DX1.4 == "9539" | inj.sub$DX1.4 ==
    "9571" | inj.sub$DX1.4 %betw1% c("9578", "9579") | inj.sub$DX1.4 %betw1%
    c("9598", "9599")] <- 35

inj.sub$ISRSITE[inj.sub$DX1.3 %betw1% c("930", "939") | inj.sub$DX1.3 %betw1%
    c("960", "994") | inj.sub$DX1.3 %betw1% c("905", "908") |
    inj.sub$DX1.4 %betw1% c("9090", "9092") | inj.sub$DX1.3 ==
    "958" | inj.sub$DX1.5 %betw1% c("99554", "99550") | inj.sub$DX1.5 ==
    "99559" | inj.sub$DX1.4 == "9094" | inj.sub$DX1.4 == "9099" |
    inj.sub$DX1.5 %betw1% c("99580", "99585")] <- 36

table(inj.sub$ISRSITE)


# ISRSITE2
inj.sub$ISRSITE2[(inj.sub$ISRSITE >= 1 & inj.sub$ISRSITE <= 3)] <- 1
inj.sub$ISRSITE2[(inj.sub$ISRSITE >= 4 & inj.sub$ISRSITE <= 8)] <- 2
inj.sub$ISRSITE2[(inj.sub$ISRSITE >= 9 & inj.sub$ISRSITE <= 13)] <- 3
inj.sub$ISRSITE2[(inj.sub$ISRSITE >= 14 & inj.sub$ISRSITE <=
    18)] <- 4
inj.sub$ISRSITE2[(inj.sub$ISRSITE >= 19 & inj.sub$ISRSITE <=
    23)] <- 5
inj.sub$ISRSITE2[(inj.sub$ISRSITE >= 24 & inj.sub$ISRSITE <=
    27)] <- 6
inj.sub$ISRSITE2[(inj.sub$ISRSITE >= 28 & inj.sub$ISRSITE <=
    33)] <- 7
inj.sub$ISRSITE2[(inj.sub$ISRSITE >= 34 & inj.sub$ISRSITE <=
    35)] <- 8
inj.sub$ISRSITE2[inj.sub$ISRSITE == 36] <- 9
```

```r
# ISRSITE3
inj.sub$ISRSITE3[(inj.sub$ISRSITE >= 1 & inj.sub$ISRSITE <= 8)] <- 1
inj.sub$ISRSITE3[(inj.sub$ISRSITE >= 9 & inj.sub$ISRSITE <= 18)] <- 2
inj.sub$ISRSITE3[(inj.sub$ISRSITE >= 19 & inj.sub$ISRSITE <=
    23)] <- 3
inj.sub$ISRSITE3[(inj.sub$ISRSITE >= 24 & inj.sub$ISRSITE <=
    33)] <- 4
inj.sub$ISRSITE3[(inj.sub$ISRSITE >= 34 & inj.sub$ISRSITE <=
    36)] <- 5


# create descriptive names for the codes above (NB: removed '
# & LATE EFFECTS' from 'SYSTEM WIDE' because those icd9 codes
# were removed when creating the data set)

# descriptors for ISRCODE
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 1] <- "FRACTURES "
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 2] <- "DISLOCATION"
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 3] <- "SPRAINS&STRAINS"
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 4] <- "INTERNAL ORGAN "
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 5] <- "OPEN WOUNDS"
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 6] <- "AMPUTATIONS"
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 7] <- "BLOOD VESSELS"
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 8] <- "SUPERFIC/CONT"
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 9] <- "CRUSHING"
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 10] <- "BURNS"
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 11] <- "NERVES"
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 12] <- "UNSPECIFIED"
inj.sub$ISRCODE.descr[inj.sub$ISRCODE == 13] <- "SYSTEM WIDE"


# descriptors for ISRSITE
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 1] <- "TYPE 1 TBI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 2] <- "TYPE 2 TBI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 3] <- "TYPE 3 TBI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 4] <- "OTHER HEAD"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 5] <- "FACE"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 6] <- "EYE"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 7] <- "NECK"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 8] <- "HEAD,FACE,NECK UNSPEC"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 9] <- "CERVICAL SCI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 10] <- "THORACIC/DORSAL SCI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 11] <- "LUMBAR SCI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 12] <- "SACRUM COCCYX SCI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 13] <- "SPINE+BACK UNSPEC SCI"
```

```r
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 14] <- "CERVICAL VCI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 15] <- "THORACIC/DORSAL VCI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 16] <- "LUMBAR VCI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 17] <- "SACRUM COCCYX VCI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 18] <- "SPINE,BACK UNSPEC VCI"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 19] <- "CHEST"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 20] <- "ABDOMEN"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 21] <- "PELVIS+UROGENITAL"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 22] <- "TRUNK"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 23] <- "BACK+BUTTOCK"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 24] <- "SHOULDER&UPPER ARM"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 25] <- "FOREARM&ELBOW"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 26] <- "HAND&WRIST&FINGERS"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 27] <- "OTHER&UNSPEC UPPER EXTREM"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 28] <- "HIP"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 29] <- "UPPER LEG&THIGH"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 30] <- "KNEE"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 31] <- "LOWER LEG&ANKLE"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 32] <- "FOOT&TOES"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 33] <- "OTHER&UNSPEC LOWER EXTREM"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 34] <- "OTHER,MULTIPLE,NEC"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 35] <- "UNSPECIFIED"
inj.sub$ISRSITE.descr[inj.sub$ISRSITE == 36] <- "SYSTEM WIDE"


# descriptors for ISRSITE2
inj.sub$ISRSITE2.descr[inj.sub$ISRSITE2 == 1] <- "TBI"
inj.sub$ISRSITE2.descr[inj.sub$ISRSITE2 == 2] <- "OTH HEAD,FACE,NECK"
inj.sub$ISRSITE2.descr[inj.sub$ISRSITE2 == 3] <- "SCI"
inj.sub$ISRSITE2.descr[inj.sub$ISRSITE2 == 4] <- "VCI "
inj.sub$ISRSITE2.descr[inj.sub$ISRSITE2 == 5] <- "TORSO"
inj.sub$ISRSITE2.descr[inj.sub$ISRSITE2 == 6] <- "UPPER EXTREMITY"
inj.sub$ISRSITE2.descr[inj.sub$ISRSITE2 == 7] <- "LOWER EXTREMITY"
inj.sub$ISRSITE2.descr[inj.sub$ISRSITE2 == 8] <- "OTHER & UNSPECIFIED"
inj.sub$ISRSITE2.descr[inj.sub$ISRSITE2 == 9] <- "SYSTEM WIDE"

# descriptors for ISRSITE3
inj.sub$ISRSITE3.descr[inj.sub$ISRSITE3 == 1] <- "HEAD&NECK"
inj.sub$ISRSITE3.descr[inj.sub$ISRSITE3 == 2] <- "SPINE&BACK"
inj.sub$ISRSITE3.descr[inj.sub$ISRSITE3 == 3] <- "TORSO"
inj.sub$ISRSITE3.descr[inj.sub$ISRSITE3 == 4] <- "EXTREMITIES"
inj.sub$ISRSITE3.descr[inj.sub$ISRSITE3 == 5] <- "UNCLASSIFIABLE BY SITE"


table(inj.sub$ISRSITE3.descr)
table(inj.sub$ISRSITE2.descr)
table(inj.sub$ISRSITE.descr)
```

```r
table(inj.sub$ISRCODE.descr)

table(inj.sub$ISRCODE.descr, inj.sub$ISRSITE2.descr)

saveRDS(inj.sub, "~/nis.00_11.inj.rds")
```

### 4.4 Calculate and Add ICD-Derived Injury Severity Scores (ICISS)

The ICD-derived ISS (ICISS) was proposed by Osler et al in 1996. It is based on so-called "survival risk ratios" which are "...calculated as the ratio of the number of times of a given ICD-9 code occurs in a surviving patient to the total number of occurances of that code. The ICISS is defined as the product of all the survival risk ratios for each of an individual patient's injuries (for as many as ten different injuries)" (M. Segui Gomez, p291, in Li and Baker) Severe injuries have been posited as those below 0.94 (see http://www.cdc.gov/nchs/injury/ice/boston2009/boston2009_proceedings.htm#proceeding_20) and have also been categorized as Minor (ICISS 0.941 to 1.0), Moderate (ICISS 0.665 to 0.940), Serious (ICISS 0.355 to 0.664), Severe (ICISS 0.220 to 0. 354), and Critical (ICISS 0 to 0.219) (Ordered risk categories for the ICD-based injury severity score (ICISS) http://www.cdc.gov/nchs/injury/ice/boston2009/boston2009_proceedings.htm#proceeding_20)

I initially used the primary injury dx's to calculate the srr's, but later used a different approach that included all diagnoses.

```r
inj.sub <- inj[, c(19, 23:37, 67, 75), with = FALSE]  # with=FALSE index data.table columns like data f
rm(inj)

# inj.sub<-inj[sample(nrow(inj),10000),]

injury.codes <- as.character(c(800:996, 8000:9960, 80000:99600))

lateEffects <- as.character(c(9050:9099))
insectBites <- as.character(c(9104, 9105, 9114, 9115, 9124, 9125,
    9134, 9135, 9144, 9145, 9154, 9155, 9164, 9165, 9174, 9175,
    9194, 9195))
poisoning <- as.character(c(9600:9649, 96500:96502, 96509, 9651,
    9654, 9655, 96561, 96569, 9657:9690, 96900:96909, 96970:96973,
    9691:9697, 9670:9679, 96979, 9698:9809, 97081, 97089, 981,
    9820:9859, 986, 9870:9897, 98981:98989, 9899, 990, 9910:9952,
    99520:99529, 9953, 9954))
anaphylaxis <- as.character(c(99560:99569, 9957))
misc <- as.character(c(99586:99600))
non.trauma <- c(lateEffects, insectBites, poisoning, anaphylaxis,
    misc)


injury.codes <- injury.codes[!injury.codes %in% non.trauma]
```

```r
# create trauma diagnoses from which to calculate srr's
inj.sub$DX2.T <- ifelse(inj.sub$DX2 %in% injury.codes, inj.sub$DX2,
    NA)
inj.sub$DX3.T <- ifelse(inj.sub$DX3 %in% injury.codes, inj.sub$DX3,
    NA)
inj.sub$DX4.T <- ifelse(inj.sub$DX4 %in% injury.codes, inj.sub$DX4,
    NA)
inj.sub$DX5.T <- ifelse(inj.sub$DX5 %in% injury.codes, inj.sub$DX5,
    NA)
inj.sub$DX6.T <- ifelse(inj.sub$DX6 %in% injury.codes, inj.sub$DX6,
    NA)
inj.sub$DX7.T <- ifelse(inj.sub$DX7 %in% injury.codes, inj.sub$DX7,
    NA)
inj.sub$DX8.T <- ifelse(inj.sub$DX8 %in% injury.codes, inj.sub$DX8,
    NA)
inj.sub$DX9.T <- ifelse(inj.sub$DX9 %in% injury.codes, inj.sub$DX9,
    NA)
inj.sub$DX10.T <- ifelse(inj.sub$DX10 %in% injury.codes, inj.sub$DX10,
    NA)
inj.sub$DX11.T <- ifelse(inj.sub$DX11 %in% injury.codes, inj.sub$DX11,
    NA)
inj.sub$DX12.T <- ifelse(inj.sub$DX12 %in% injury.codes, inj.sub$DX12,
    NA)
inj.sub$DX13.T <- ifelse(inj.sub$DX13 %in% injury.codes, inj.sub$DX13,
    NA)
inj.sub$DX14.T <- ifelse(inj.sub$DX14 %in% injury.codes, inj.sub$DX14,
    NA)
inj.sub$DX15.T <- ifelse(inj.sub$DX15 %in% injury.codes, inj.sub$DX15,
    NA)


# calculate srr's from DX1
srrDx1.1 <- as.data.frame(table(inj.sub$DX1[inj.sub$DIED_VISIT ==
    0]))  # use table() to count up all patients who survived that dx
srrDx1.2 <- as.data.frame(table(inj.sub$DX1))  # count up all patients with the diagnosis
srrDx1.3 <- merge(srrDx1.1, srrDx1.2, by = "Var1", all = TRUE)  # create single file,
# first column the dx, second the number of surviviors, third
# the total

names(srrDx1.3) <- c("ICD", "Surv", "Tot")  # rename
srrDx1.3$Surv[is.na(srrDx1.3$Surv)] <- 0  # set NA's to zero to allow calculations
# srrDx1.3$srr<-srrDx1.3$Surv/srrDx1.3$Tot # calculate the
# srr for the dx srrDx1.3<-srrDx1.3[,-c(2,3)] # remove the
# first two columns of counts

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

```r
# calculate srr's from DX2
srrDx2.1 <- as.data.frame(table(inj.sub$DX2.T[inj.sub$DIED_VISIT ==
    0]))
srrDx2.2 <- as.data.frame(table(inj.sub$DX2.T))
srrDx2.3 <- merge(srrDx2.1, srrDx2.2, by = "Var1", all = TRUE)

names(srrDx2.3) <- c("ICD", "Surv", "Tot")
srrDx2.3$Surv[is.na(srrDx2.3$Surv)] <- 0
# srrDx2.3$srr<-srrDx2.3$Surv/srrDx2.3$Tot
# srrDx2.3<-srrDx2.3[,-c(2,3)]

#############################################

# calculate srr's from DX3
srrDx3.1 <- as.data.frame(table(inj.sub$DX3.T[inj.sub$DIED_VISIT ==
    0]))
srrDx3.2 <- as.data.frame(table(inj.sub$DX3.T))
srrDx3.3 <- merge(srrDx3.1, srrDx3.2, by = "Var1", all = TRUE)

names(srrDx3.3) <- c("ICD", "Surv", "Tot")
srrDx3.3$Surv[is.na(srrDx3.3$Surv)] <- 0
# srrDx3.3$srr<-srrDx3.3$Surv/srrDx3.3$Tot
# srrDx3.3<-srrDx3.3[,-c(2,3)]

############################################# calculate srr's from DX4
srrDx4.1 <- as.data.frame(table(inj.sub$DX4.T[inj.sub$DIED_VISIT ==
    0]))
srrDx4.2 <- as.data.frame(table(inj.sub$DX4.T))
srrDx4.3 <- merge(srrDx4.1, srrDx4.2, by = "Var1", all = TRUE)

names(srrDx4.3) <- c("ICD", "Surv", "Tot")
srrDx4.3$Surv[is.na(srrDx4.3$Surv)] <- 0
# srrDx4.3$srr<-srrDx4.3$Surv/srrDx4.3$Tot
# srrDx4.3<-srrDx4.3[,-c(2,3)]

#############################################

# calculate srr's from DX5
srrDx5.1 <- as.data.frame(table(inj.sub$DX5.T[inj.sub$DIED_VISIT ==
    0]))
srrDx5.2 <- as.data.frame(table(inj.sub$DX5.T))
srrDx5.3 <- merge(srrDx5.1, srrDx5.2, by = "Var1", all = TRUE)

names(srrDx5.3) <- c("ICD", "Surv", "Tot")
srrDx5.3$Surv[is.na(srrDx5.3$Surv)] <- 0
# srrDx5.3$srr<-srrDx5.3$Surv/srrDx5.3$Tot
```

```r
# srrDx5.3<-srrDx5.3[,-c(2,3)]

#############################################


# calculate srr's from DX6
srrDx6.1 <- as.data.frame(table(inj.sub$DX6.T[inj.sub$DIED_VISIT ==
    0]))
srrDx6.2 <- as.data.frame(table(inj.sub$DX6.T))
srrDx6.3 <- merge(srrDx6.1, srrDx6.2, by = "Var1", all = TRUE)

names(srrDx6.3) <- c("ICD", "Surv", "Tot")
srrDx6.3$Surv[is.na(srrDx6.3$Surv)] <- 0
# srrDx6.3$srr<-srrDx6.3$Surv/srrDx6.3$Tot
# srrDx6.3<-srrDx6.3[,-c(2,3)]

#############################################

# calculate srr's from DX7
srrDx7.1 <- as.data.frame(table(inj.sub$DX7.T[inj.sub$DIED_VISIT ==
    0]))
srrDx7.2 <- as.data.frame(table(inj.sub$DX7.T))
srrDx7.3 <- merge(srrDx7.1, srrDx7.2, by = "Var1", all = TRUE)

names(srrDx7.3) <- c("ICD", "Surv", "Tot")
srrDx7.3$Surv[is.na(srrDx7.3$Surv)] <- 0
# srrDx7.3$srr<-srrDx7.3$Surv/srrDx7.3$Tot
# srrDx7.3<-srrDx7.3[,-c(2,3)]

#############################################

# calculate srr's from DX8
srrDx8.1 <- as.data.frame(table(inj.sub$DX8.T[inj.sub$DIED_VISIT ==
    0]))
srrDx8.2 <- as.data.frame(table(inj.sub$DX8.T))
srrDx8.3 <- merge(srrDx8.1, srrDx8.2, by = "Var1", all = TRUE)

names(srrDx8.3) <- c("ICD", "Surv", "Tot")
srrDx8.3$Surv[is.na(srrDx8.3$Surv)] <- 0
# srrDx8.3$srr<-srrDx8.3$Surv/srrDx8.3$Tot
# srrDx8.3<-srrDx8.3[,-c(2,3)]

#############################################

# calculate srr's from DX9
srrDx9.1 <- as.data.frame(table(inj.sub$DX9.T[inj.sub$DIED_VISIT ==
    0]))
```

```r
srrDx9.2 <- as.data.frame(table(inj.sub$DX9.T))
srrDx9.3 <- merge(srrDx9.1, srrDx9.2, by = "Var1", all = TRUE)

names(srrDx9.3) <- c("ICD", "Surv", "Tot")
srrDx9.3$Surv[is.na(srrDx9.3$Surv)] <- 0
# srrDx9.3$srr<-srrDx9.3$Surv/srrDx9.3$Tot
# srrDx9.3<-srrDx9.3[,-c(2,3)]

#############################################

# calculate srr's from DX10
srrDx10.1 <- as.data.frame(table(inj.sub$DX10.T[inj.sub$DIED_VISIT ==
    0]))
srrDx10.2 <- as.data.frame(table(inj.sub$DX10.T))
srrDx10.3 <- merge(srrDx10.1, srrDx10.2, by = "Var1", all = TRUE)

names(srrDx10.3) <- c("ICD", "Surv", "Tot")
srrDx10.3$Surv[is.na(srrDx10.3$Surv)] <- 0
# srrDx10.3$srr<-srrDx10.3$Surv/srrDx10.3$Tot
# srrDx10.3<-srrDx10.3[,-c(2,3)]

#############################################

# calculate srr's from DX10
srrDx11.1 <- as.data.frame(table(inj.sub$DX11.T[inj.sub$DIED_VISIT ==
    0]))
srrDx11.2 <- as.data.frame(table(inj.sub$DX11.T))
srrDx11.3 <- merge(srrDx11.1, srrDx11.2, by = "Var1", all = TRUE)

names(srrDx11.3) <- c("ICD", "Surv", "Tot")
srrDx11.3$Surv[is.na(srrDx11.3$Surv)] <- 0
# srrDx10.3$srr<-srrDx10.3$Surv/srrDx10.3$Tot
# srrDx10.3<-srrDx10.3[,-c(2,3)]

#############################################

# calculate srr's from DX10
srrDx12.1 <- as.data.frame(table(inj.sub$DX12.T[inj.sub$DIED_VISIT ==
    0]))
srrDx12.2 <- as.data.frame(table(inj.sub$DX12.T))
srrDx12.3 <- merge(srrDx12.1, srrDx12.2, by = "Var1", all = TRUE)

names(srrDx12.3) <- c("ICD", "Surv", "Tot")
srrDx12.3$Surv[is.na(srrDx12.3$Surv)] <- 0
# srrDx10.3$srr<-srrDx10.3$Surv/srrDx10.3$Tot
# srrDx10.3<-srrDx10.3[,-c(2,3)]
```

```r
#############################################

# calculate srr's from DX10
srrDx13.1 <- as.data.frame(table(inj.sub$DX13.T[inj.sub$DIED_VISIT ==
    0]))
srrDx13.2 <- as.data.frame(table(inj.sub$DX13.T))
srrDx13.3 <- merge(srrDx13.1, srrDx13.2, by = "Var1", all = TRUE)

names(srrDx13.3) <- c("ICD", "Surv", "Tot")
srrDx13.3$Surv[is.na(srrDx13.3$Surv)] <- 0
# srrDx10.3$srr<-srrDx10.3$Surv/srrDx10.3$Tot
# srrDx10.3<-srrDx10.3[,-c(2,3)]

#############################################

# calculate srr's from DX10
srrDx14.1 <- as.data.frame(table(inj.sub$DX14.T[inj.sub$DIED_VISIT ==
    0]))
srrDx14.2 <- as.data.frame(table(inj.sub$DX14.T))
srrDx14.3 <- merge(srrDx14.1, srrDx14.2, by = "Var1", all = TRUE)

names(srrDx14.3) <- c("ICD", "Surv", "Tot")
srrDx14.3$Surv[is.na(srrDx14.3$Surv)] <- 0
# srrDx10.3$srr<-srrDx10.3$Surv/srrDx10.3$Tot
# srrDx10.3<-srrDx10.3[,-c(2,3)]

#############################################

# calculate srr's from DX10
srrDx15.1 <- as.data.frame(table(inj.sub$DX15.T[inj.sub$DIED_VISIT ==
    0]))
srrDx15.2 <- as.data.frame(table(inj.sub$DX15.T))
srrDx15.3 <- merge(srrDx15.1, srrDx15.2, by = "Var1", all = TRUE)

names(srrDx15.3) <- c("ICD", "Surv", "Tot")
srrDx15.3$Surv[is.na(srrDx15.3$Surv)] <- 0
# srrDx10.3$srr<-srrDx10.3$Surv/srrDx10.3$Tot
# srrDx10.3<-srrDx10.3[,-c(2,3)]

#############################################



# merge all the individual dx data sets


merged.srrDxs.1 <- merge(srrDx1.3, srrDx2.3, by = "ICD", all = T)
```

```r
merged.srrDxs.2 <- merge(srrDx3.3, srrDx4.3, by = "ICD", all = T)
merged.srrDxs.3 <- merge(srrDx5.3, srrDx6.3, by = "ICD", all = T)
merged.srrDxs.4 <- merge(srrDx7.3, srrDx8.3, by = "ICD", all = T)
merged.srrDxs.5 <- merge(srrDx9.3, srrDx10.3, by = "ICD", all = T)
merged.srrDxs.6 <- merge(srrDx11.3, srrDx12.3, by = "ICD", all = T)
merged.srrDxs.7 <- merge(srrDx13.3, srrDx14.3, by = "ICD", all = T)
merged.srrDxs.8 <- srrDx15.3


merged.srrDxs.9 <- merge(merged.srrDxs.1, merged.srrDxs.2, by = "ICD",
    all = T)
merged.srrDxs.10 <- merge(merged.srrDxs.3, merged.srrDxs.4, by = "ICD",
    all = T)
merged.srrDxs.11 <- merge(merged.srrDxs.5, merged.srrDxs.6, by = "ICD",
    all = T)
merged.srrDxs.12 <- merge(merged.srrDxs.7, merged.srrDxs.8, by = "ICD",
    all = T)


merged.srrDxs.13 <- merge(merged.srrDxs.9, merged.srrDxs.10,
    by = "ICD", all = T)
merged.srrDxs.14 <- merge(merged.srrDxs.11, merged.srrDxs.12,
    by = "ICD", all = T)

merged.srrDxs.15 <- merge(merged.srrDxs.13, merged.srrDxs.14,
    by = "ICD", all = T)


merged.srrDxs.15 <- replace(merged.srrDxs.15, is.na(merged.srrDxs.15),
    0)

str(merged.srrDxs.15)
merged.srrDxs.15$Tot <- rowSums(merged.srrDxs.15[, c(3, 5, 7,
    9, 11, 13, 15, 17, 19, 21)], na.rm = T)
merged.srrDxs.15$Surv <- rowSums(merged.srrDxs.15[, c(2, 4, 6,
    8, 10, 12, 14, 16, 18, 20)], na.rm = T)
merged.srrDxs.15$srr <- merged.srrDxs.15$Surv/merged.srrDxs.15$Tot

srr <- merged.srrDxs.15[, c("ICD", "srr")]

# save the ssr file write.csv(srr,
# file='~nis.06_12.ssrs.csv')
# srr<-read.csv('~/nis.06_12.ssrs.csv',header=T,
# stringsAsFactors=F)


# tidy up workspace
```

```r
rm(anaphylaxis, injury.codes, insectBites, lateEffects, merged.srrDxs.1,
    merged.srrDxs.10, merged.srrDxs.11, merged.srrDxs.12, merged.srrDxs.13,
    merged.srrDxs.14, merged.srrDxs.15, merged.srrDxs.2, merged.srrDxs.3,
    merged.srrDxs.4, merged.srrDxs.5, merged.srrDxs.6, merged.srrDxs.7,
    merged.srrDxs.8, merged.srrDxs.9, misc, non.trauma, poisoning,
    srrDx1.1, srrDx1.2, srrDx1.3, srrDx10.1, srrDx10.2, srrDx10.3,
    srrDx11.1, srrDx11.2, srrDx11.3, srrDx12.1, srrDx12.2, srrDx12.3,
    srrDx13.1, srrDx13.2, srrDx13.3, srrDx14.1, srrDx14.2, srrDx14.3,
    srrDx15.1, srrDx15.2, srrDx15.3, srrDx2.1, srrDx2.2, srrDx2.3,
    srrDx3.1, srrDx3.2, srrDx3.3, srrDx4.1, srrDx4.2, srrDx4.3,
    srrDx5.1, srrDx5.2, srrDx5.3, srrDx6.1, srrDx6.2, srrDx6.3,
    srrDx7.1, srrDx7.2, srrDx7.3, srrDx8.1, srrDx8.2, srrDx8.3,
    srrDx9.1, srrDx9.2, srrDx9.3)

# merge the srr to the dataframe matching on the ICD code
# need as.data.frame because data.table merge doesn't
# recognize by.x and by.y
inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX1",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr1"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX2.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr2"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX3.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr3"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX4.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr4"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX5.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr5"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX6.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr6"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX7.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr7"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX8.T",
    by.y = "ICD", all.x = T)
```

```
names(inj.sub)[names(inj.sub) == "srr"] <- "srr8"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX9.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr9"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX10.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr10"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX11.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr11"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX12.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr12"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX13.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr13"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX14.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr14"

inj.sub <- merge(x = as.data.frame(inj.sub), y = srr, by.x = "DX15.T",
    by.y = "ICD", all.x = T)
names(inj.sub)[names(inj.sub) == "srr"] <- "srr15"


# calculate iciss as product of ssr's
inj.sub$iciss <- apply(inj.sub[, c("srr1", "srr2", "srr3", "srr4",
    "srr5", "srr6", "srr7", "srr8", "srr9", "srr10", "srr11",
    "srr12", "srr13", "srr14", "srr15")], 1, prod, na.rm = T)
class(inj.sub)

inj.iciss <- inj.sub[, c(31, 32, 48)]

library(dplyr)
inj.iciss <- tbl_df(inj.iciss)
inj.iciss <- arrange(inj.iciss, KEY_ED, YEAR)

inj <- tbl_df(inj)
inj <- arrange(inj, KEY_ED, YEAR)
inj2 <- merge(inj, inj.iciss, by = c("KEY_ED", "YEAR"))
saveRDS(inj2, "~/neds.06_12.inj.rds")
```

### 4.5 Calculate Charlson Index

Use icd9::icd9Charlson() to calculate Charlson score for each patient in the individual data set. First create a data frame limited to identifier and diagnostic codes. Convert this dataframe from a wide to a long format, Then apply the function [7]

```
charlDat <- data.frame(inj[, c("KEY_ED", "DX1", "DX2", "DX3",
    "DX4", "DX5", "DX6", "DX7", "DX8", "DX9", "DX10", "DX11",
    "DX12", "DX13", "DX14")])  # subset inj
rm(inj)  # remove inj to clear up memory
nrow(charlDat)  #40073358

charlDat <- data.table(charlDat)  # convert to datatable use optimized version of melt()


charlDat2 <- melt(charlDat, id.vars = "KEY_ED", na.rm = TRUE)
nrow(charlDat2)  #561027012 # wow, fast...

rm(charlDat)  # free up space

# icd9::icd9Charlson()
charlDat2 <- charlDat2[, -2, with = FALSE]  # drop variable so only 2 required variables
injCharlson <- icd9Charlson(charlDat2, visitId = "KEY_ED", return.df = TRUE)

head(injCharlson)
mean(injCharlson$Charlson)

nrow(injCharlson)  #40073358

injCharlson <- arrange(injCharlson, KEY_ED)

rm(charlDat2, checkvar, test)
# merge back to original data set

inj <- readRDS("~/neds.06_12.inj.rds")

inj2 <- merge(x = inj, y = injCharlson, by = "KEY_ED")

saveRDS(inj2, "~/neds.06_12.inj.rds")

str(inj2)
```

[7] First pass tried using R package "medicalrisk" without success. I think it didn't read the icd9 codes well.

## 4.6 Add Trauma Center Variable

HOSP_TRAUMA indicator variable is in hospital file.

```
hospNames<- c("DISCWT", "HOSPWT", "HOSP_CONTROL", "HOSP_ED", "HOSP_REGION", "HOSP_TRAUMA", "HOSP_URCA


hosp2012<-read.csv("~/2012/NEDS_2012_HOSPITAL.csv", stringsAsFactors=F,
col.names=hospNames)
hosp2011<-read.csv("~/2011/NEDS_2011_HOSPITAL.csv", stringsAsFactors=F,
col.names=hospNames)
hosp2010<-read.csv("~/2010/NEDS_2010_HOSPITAL.csv", stringsAsFactors=F,
col.names=hospNames)
hosp2009<-read.csv("~/2009/NEDS_2009_HOSPITAL.csv", stringsAsFactors=F,
col.names=hospNames)
hosp2008<-read.csv("~/2008/NEDS_2008_HOSPITAL.csv", stringsAsFactors=F,
col.names=hospNames)
hosp2007<-read.csv("~/2007/NEDS_2007_HOSPITAL.csv", stringsAsFactors=F,
col.names=hospNames)
hosp2006<-read.csv("~/2006/NEDS_2006_HOSPITAL.csv", stringsAsFactors=F,
col.names=hospNames)

hosp0612<-rbind(hosp2012,hosp2011,hosp2010,hosp2009,hosp2008,hosp2007,hosp2006)

# keep vars for hosp id, trauma, teaching, number ed visits, year
hosp0612<-hosp0612[,c(4,6,8,14,15)]
nrow(hosp0612) # 6720
length(unique(hosp0612$HOSP_ED)) #2521,  id's repeat, need to merge with id and year

inj<-tbl_df(inj)
inj<-arrange(inj, HOSP_ED, YEAR)

hosp<-tbl_df(hosp0612)
hosp<-arrange(hosp, HOSP_ED, YEAR)

inj2<-inj %>% left_join(hosp,  by=c("HOSP_ED","YEAR"))

nrow(inj2)

table(inj2$HOSP_TRAUMA)

inj<-inj2
rm(inj2)

#create indicator/count variable for later analyses

inj$count<-1
```

```
# save the file
system.time(saveRDS(inj,"~/neds.06_12.inj.rds"))
```

## 4.7 Calculate Cost Variable

Cost-to-charge files are not available for NEDS as they are for NIS. Conservative estimate is to calculate costs as 42% of charges (see Hsia RY, MacIsaac D, Baker LC. Decreasing reimbursements for outpatient emergency department visits across payer groups from 1996 to 2004. Ann Emerg Med. 2008;51:265–74. 274. cited in http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3431910/ see also http://www.usc.edu/schools/price/research/healthresearch/images/pdf_reportspapers/cost%20of%20emergency%20department%20visit.pdf)

Inflation adjusting approach is to use the average yearly consumer price index (obtained from the Bureau of Labor statistics, see http://www.usinflationcalculator.com/inflation/consumer-price-index-and-annual-percent-changes-from-1913-to-2008/), divide the average yearly CPIs by the value for the year to which you want to standardize, (here, 2010), then multiply the costs by that factor (see http://stackoverflow.com/questions/12590180/inflation-adjusted-prices-package and http://people.duke.edu/~rnau/411infla.htm) )

```
inj$TOTCHG_ED[inj$TOTCHG_ED<0]<-NA # identify NAs
summary(inj$TOTCHG_ED)
inj$tot_cost<-inj$TOTCHG_ED*0.42 # cost as 42% of charges
summary(inj$tot_cost)
mean(inj$tot_cost, na.rm=T)

# inflation adjust to 2012 dollars
cpi <- data.frame(cpi= c(201.6/229.594, 207.3/229.594, 215.303/229.594, 214.537/229.594,
        218.056/229.594, 224.939/229.594,229.594/229.594),
        YEAR=c(2006,2007,2008,2009,2010,2011,2012))
inj<-merge(x=inj, y=cpi, by="YEAR", all.x=T, all.y=F)
nrow(inj)
table(inj$cpi)
inj$adj_charge<-inj$TOTCHG_ED*inj$cpi
mean(inj$adj_charge, na.rm=T)
inj$adj_cost<-inj$tot_cost*inj$cpi
mean(inj$adj_cost, na.rm=T)

#clean up and save inj file
names(inj)[-c(65,66)]
head(inj[,-c(65,66)])

inj<-inj[,-c(65,66)]
system.time(saveRDS(inj,"~/neds.06_12.inj.rds"))
```

Create indicator and categorical variables for mortality, injury severity, Charlson score, and level 1 or 2 trauma center status. Create a count variable for use with survey procedures.

```
inj$died<-1
inj$died[inj$DIED_VISIT==0]<-0
inj<-readRDS("/Volumes/Promise Pegasus/DATA/HCUP/NEDSInjury/neds.06_12.inj.rds")
inj$count<-1
inj$severe<-0
inj$severe[inj$iciss< 0.941]<-1
inj$charleson_severe<-0
inj$charleson_severe[inj$Charlson > 2]<-1
inj$traumaCenter<-0
inj$traumaCenter[inj$HOSP_TRAUMA==1 | inj$HOSP_TRAUMA==2]<-1
inj$traumaCenter[inj$HOSP_TRAUMA==8 | inj$HOSP_TRAUMA==9]<-NA
# names(inj)
saveRDS(inj, file="/~/neds.06_12.inj.rds")
```

## 4.8 Write MonetDB File

Large file running up against R memory limits will tax even the most robust machines. One solution is to use an out-of-memory file. MonetDB is a column-based database system that can retrieve results relatively quickly, can interface with R with the "MonetDBLite" and can be analyzed with the "sqlsurvey" package which is based on "survey".[8]

Begin by cleaning up and reducing the existing R file.

```
names(inj)

object.size(inj)/2e30 # 1.87 GB

# only first chronic disease indicator variable has full data
sum(inj$CHRON1 %in% c(-99,-88,-66))/nrow(inj)*100 # missing 0%
sum(inj$CHRON2 %in% c(-99,-88,-66))/nrow(inj)*100 # missing 45%
sum(inj$CHRON6 %in% c(-99,-88,-66))/nrow(inj)*100 # missing 91%


# remove chronic disease indicator variables 2 through 15
inj<-inj[,-c(7:20)]

# remove dx clinical classification software variables; useful for overall dx's, but are duplicated by
inj<-inj[,-c(26:40)]

# remove additional dx's; useful for creating Charlson score, but not necessary for primary analyses
inj<-inj[,-c(12:25)]
```

[8]A description of working with large HCUP files using MonetDB and plyr can be found at http://www.injuryepi.org/resources/Misc/hcupNotesMonet.pdf

```r
# remove the 3, 4 and 5-digit versions of ICD9 codes used to create Barrel matrix
inj<-inj[,-c(53:56)]

# remove number of ecodes and number of diagnoses variables
inj<-inj[,-c(26:27)]
saveRDS(inj, file="/~/neds.06_12.inj.rds")
```

Use system tools to create " /monetInjury" folder to hold the files on hard drive.[9] Fix the variable names by converting to lower case and renaming reserved variable names. Create a database connection to the folder that will hold the MonetDB file using DBI::dbConnect(), specifying that it will be a MonetDBLite connection. Use DBI::dbWriteTable() to convert the R file to a MonetDB file.

```r
mdb <- dbConnect(MonetDBLite(), "~/monetInjury")
# fix variable names for monet
names(inj)[c(1,64,50,55:58)]<-c("yr","inj","agegrp_num" ,"isrcode_descr", "isrsite_descr","isrsite2_
names(inj)<-tolower(names(inj))
names(inj)
dbWriteTable(mdb, "nedsinj0612", inj) # write the injury file to the monetdb database
dbListTables(mdb)
dbGetInfo(mdb)
dbListFields(mdb,"nedsinj0612")
```

Use SQL to update the MonetDB file to create and id variable that is required by "sqlsurvey" If not done before writing the R file to MonetDB, create indicator and categorical variables for mortality, teaching, charlson, and trauma center.

```r
mdb <- dbConnect(MonetDBLite(), "/Volumes/Promise Pegasus/DATA/HCUP/NEDSInjury/monetInjury")
# create an id variable (required by sqlsurvey), a count variable, and dichotomized severity variable
# dbSendQuery(mdb,"alter table nedsinj0612 add id serial;")
# dbSendQuery(mdb,"alter table nedsinj0612 add count int default 1;")
# dbSendQuery(mdb,"alter table nedsinj0612 add severe int default 0;")
# dbSendQuery(mdb,"update nedsinj0612 set severe=1 where iciss < 0.941;")

# create dichotomous charleson score
# dbSendQuery(mdb,"alter table nedsinj0612 add charleson_severe int default 0;")
# dbSendQuery(mdb,"update nedsinj0612 set charleson_severe=1 where charlson > 2;")

# dichotmous teaching variable
# dbSendQuery(mdb,"alter table nedsinj0612 add teach int default 1;")
# dbSendQuery(mdb,"update nedsinj0612 set teach=0 where hosp_ur_teach = 0;")


# dichotmous death variable
# dbSendQuery(mdb,"alter table nedsinj0612 add died int default 1;")
# dbSendQuery(mdb,"update nedsinj0612 set died=0 where died_visit = 0;")
```

[9]Ideally not a hard drive that holds the system operating system or R.

```
# dichotmous trauma center variable
# dbSendQuery(mdb,"alter table nedsinj0612 add trauma_center int default 0;")
# dbSendQuery(mdb,"update nedsinj0612 set trauma_center=1 where hosp_trauma = 1;")
# dbSendQuery(mdb,"update nedsinj0612 set trauma_center=1 where hosp_trauma = 2;")
# dbSendQuery(mdb,"update nedsinj0612 set trauma_center=null where hosp_trauma = 8;")
# dbSendQuery(mdb,"update nedsinj0612 set trauma_center=null where hosp_trauma = 9;")
```

# 5 Appendix: Survey Procedures Run on Full R Data Set

The following procedures were run on the full R data set before switching to "sqlsurvey" and MonetDB. They take a *very* long time to run. Calculating the mean age, for example, took nearly 4 hours on a 64GB machine and a bit less than an hour on a 128GB machine.

```
# create survey design from full R data set
svydes<- svydesign(
        id = ~key_ed ,
        strata = ~interaction(neds_stratum , yr),  # note YEAR interaction
        weights = ~discwt ,
        nest = TRUE,
        data = inj,
        multicore=T
)

save(svydes , file = "/~/svydes.rda" )
```

```
load("~/svydes.rda")


#64GB RAM interaction year
system.time(meanAGE<-svymean(~age, svydes, na.rm=T))
meanAGE #
confint(meanAGE)

#       user     system   elapsed
# 11693.648   655.212 12332.583   # 3.42 hours
# > meanAGE #
#        mean     SE
# age 34.746 0.0039
# > confint(meanAGE)
#         2.5 %   97.5 %
# age 34.73852 34.75367
```

```r
# 128 GB RAM no year interaction
system.time(meanAGE<-svymean(~age, svydes2, na.rm=T))
meanAGE #
confint(meanAGE)
#     user   system  elapsed
# 3082.131  143.628 3208.822 # bit less than an hour...
# > meanAGE #
#       mean     SE
# age 34.746 0.0039 # exact same answer...
# > confint(meanAGE)
#       2.5 %   97.5 %
# age 34.7385 34.75369


# #COMPARE TO RAW DATA
# system.time(print(mean(inj$AGE, na.rm=T)))
# # [1] 34.77108
# #    user  system elapsed
# #   0.407   0.249   0.653
#
#
# system.time(print(sd(inj$AGE, na.rm=T)))
# # [1] 23.46335
# #    user  system elapsed
# #   0.258   0.146   0.403
#
# sterr <- function(x) sd(x, na.rm=T)/sqrt(length(x))
# # system.time(print(sterr(inj$AGE)))
# # [1] 0.003706483
# #    user  system elapsed
# #   0.257   0.139   0.394

system.time(AGEbyYear<-svyby(~age, ~yr, svydes, svymean, na.rm=T, vartype = c( 'ci' , 'se' )))
save(meanAGE, AGEbyYR, file="/Users/charlie/Box Sync/hcup/NEDS/nedsInjury/ageResults.RData")
AGEbyYear
system.time(AGEbyYear<-svyby(~age, ~yr, db_design, svymean, na.rm=T, vartype = c( 'ci' , 'se' )))
#     user   system  elapsed
# 4600.050  376.661 6594.196
(6594.196 /60)/60 # 1.83 hours
AGEbyYear
#        yr      age          se     ci_l     ci_u
# 2006 2006 33.83112 0.009939669 33.81163 33.85060
# 2007 2007 34.07261 0.010055909 34.05290 34.09232
# 2008 2008 34.57061 0.009968646 34.55107 34.59014
# 2009 2009 34.87537 0.010577461 34.85464 34.89610
# 2010 2010 35.31072 0.010465413 35.29021 35.33124
# 2011 2011 35.33135 0.010312395 35.31114 35.35157
```

```
# 2012 2012 35.30092 0.010313871 35.28071 35.32114


#COMPARE TO RAW DATA
# save(meanAGE, AGEbyYear, file="/Users/charlie/Box Sync/hcup/NEDS/nedsInjury/ageResults.RData")
#
# system.time(print(tapply(inj$AGE, inj$YEAR, mean, na.rm=T)))
# #     2006     2007     2008     2009     2010     2011     2012
# # 33.86900 34.08656 34.60711 34.81538 35.27819 35.36932 35.38931
# #    user  system elapsed
# #   3.388   1.166   4.529
#
#
# system.time(print(tapply(inj$AGE, inj$YEAR, sterr)))
# #        2006        2007        2008        2009        2010        2011        2012
# # 0.009577755 0.009620235 0.009565588 0.009936695 0.009906659 0.010148218 0.009880995
# #    user  system elapsed
# #   3.237   0.990   4.186



system.time(AGEbyYRreg <- svyglm(age ~ female + yr, design=svydes))
summary(AGEbyYRreg)
confint(AGEbyYRreg)

#      user    system   elapsed
# 10000.300   653.442 10598.284
# > summary(AGEbyYRreg)
#
# Call:
# svyglm(formula = age ~ female + yr, design = svydes)
#
# Survey design:
# svydesign(id = ~key_ed, strata = ~interaction(neds_stratum, yr),
#     weights = ~discwt, nest = TRUE, data = inj, multicore = T)
#
# Coefficients:
#               Estimate Std. Error t value Pr(>|t|)
# (Intercept) -4.689e+02  3.812e+00  -123.0   <2e-16 ***
# female       6.637e+00  9.318e-03   712.2   <2e-16 ***
# yr           2.492e-01  1.898e-03   131.3   <2e-16 ***
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
# (Dispersion parameter for gaussian family taken to be 538.0092)
#
# Number of Fisher Scoring iterations: 2
```

```
#
# > confint(AGEbyYRreg)
#                    2.5 %       97.5 %
# (Intercept) -476.3598100 -461.4173291
# female         6.6184008    6.6549286
# yr             0.2454439    0.2528821
```

```
# total cost

system.time(print(svytotal(~adj_cost, svydes, na.rm=T)))
#               total       SE
# adj_cost 9.975e+10 26685092
#      user     system   elapsed
# 10005.837   610.701 10577.755

# compare to raw data approach

# SurvTot<-function(x){
#  N <- sum(1/svydes$prob)
#  m <- mean(x, na.rm = T)
#  total <- m * N
#  return(total)
# }

system.time(print(SurvTot(inj$adj_cost)))
# [1] 1.18511e+11
#    user  system elapsed
#   0.735   0.311   0.989

system.time(COSTbyYear<-svyby(~adj_cost, ~yr, svydes, svymean, na.rm=T, vartype = c( 'ci' , 'se' )))
COSTbyYear

#     cost     se     ci_l ci_u
# 2006 424.593     0.287 424.03 424.16
# 2007 495.4825 0.339 494.82 496.15
# 2008 580.3209 0.39 579.56 581.09
# 2009 638.8913 0.437 638.04 639.75
# 2010 713.5966 0.464 712.69 714.51
# 2011 786.0859 0.544 785.02 787.15
# 2012 873.0273 0.606 871.84 874.21

system.time(COSTbyDx<-svyby(~adj_cost, ~dx1, svydes, svymean, na.rm=T, vartype = c( 'ci' , 'se' )))
# Error: $ operator is invalid for atomic vectors
# In addition: Warning message:
# In (if (multicore) mclapply else lapply)(uniques, function(i) { :
#   all scheduled cores encountered errors in user code
# Timing stopped at: 47360.75 2974.117 52922.68
```

```
(52922.68/60)/60 # stopped after almost 15 hours...

# ten most frequent diagnoses
dxs<-as.data.frame(sort(table(inj$dx1), decreasing=T)[1:10])

# most expensive diagnoses
expensiveDxs<-as.data.frame(sort(tapply(inj$adj_cost,inj$dx1,mean),  decreasing=T))[1:10,]






# average cost 10 most frequent diagnoses
dxCost<-svyby(~adj_cost, ~dx1, subset(svydes, dx1 %in% rownames(dxs)), svymean, na.rm=T, vartype = c(
#    user    system   elapsed
# 109.496    29.694 13820.519
dxCost
#         dx1  adj_cost        se      ci_l      ci_u
# 8449   8449  436.8201 0.4766711  435.8858  437.7543
# 84500 84500  433.4812 0.3051651  432.8830  434.0793
# 8470   8470  801.4461 0.8465468  799.7869  803.1053
# 8472   8472  465.6632 0.6366844  464.4153  466.9111
# 8730   8730  786.3268 1.3577417  783.6657  788.9879
# 87342 87342  675.4546 1.0785952  673.3406  677.5686
# 8820   8820  426.5597 0.7135185  425.1612  427.9582
# 8830   8830  372.5561 0.3055899  371.9571  373.1550
# 920     920  852.0335 0.9072808  850.2552  853.8117
# 95901 95901 1107.9382 1.1641591 1105.6565 1110.2199

# total cost top ten most frequent diagnoses
system.time(dxTotCosts<-svyby(~adj_cost, ~dx1, subset(svydes, dx1 %in% rownames(dxs), select= c("adj_
dxTotCosts
#    user    system   elapsed
# 108.275    31.475 14361.208
# > dxTotCosts
#         dx1   adj_cost      se       ci_l       ci_u
# 8449   8449 1498530231 2394019 1493838041 1503222422
# 84500 84500 2827514797 3085846 2821466650 2833562944
# 8470   8470 5838335298 7711524 5823220988 5853449607
# 8472   8472 2364346968 3957576 2356590262 2372103674
# 8730   8730 2557131866 5386830 2546573874 2567689859
# 87342 87342 2649792857 5125574 2639746915 2659838798
# 8820   8820 1306872991 2718860 1301544122 1312201859
# 8830   8830 2627524210 3022626 2621599972 2633448449
# 920     920 5376155444 7314156 5361819961 5390490926
# 95901 95901 6066082336 8439189 6049541829 6082622843
```

```
gc()


# average cost most expensive diagnoses
system.time(avgExpensDxs<-svyby(~adj_cost, ~dx1, subset(svydes, dx1 %in% rownames(expensiveDxs), sele
avgExpensDxs
#       user    system   elapsed
# 26851.302  2164.270   276.472
# > avgExpensDxs
#          dx1 adj_cost           se       ci_l      ci_u
# 80343 80343 5332.114 2.490495e+03   450.8328 10213.395
# 80353 80353 4002.267 6.095654e+02  2807.5407  5196.993
# 80473 80473 3850.481 1.507380e+03   896.0716  6804.891
# 80483 80483 3956.723 2.264291e+03  -481.2058  8394.652
# 83511 83511 4430.500 0.000000e+00  4430.5001  4430.500
# 85233 85233 4119.218 0.000000e+00  4119.2180  4119.218
# 8975   8975 6493.240 4.556878e+03 -2438.0762 15424.557
# 94259 94259 7602.000 0.000000e+00  7602.0000  7602.000
# 94860 94860 5359.609 9.094947e-13  5359.6094  5359.609
# 9517   9517 3558.539 1.408719e+03   797.5001  6319.578


# total cost most expensive diagnoses
system.time(totExpensDxs<-svyby(~adj_cost, ~dx1, subset(svydes, dx1 %in% rownames(expensiveDxs), sele
totExpensDxs

#    user  system elapsed
# 464.272  79.056 277.491
# > totExpensDxs
#          dx1   adj_cost         se       ci_l      ci_u
# 80343 80343  50369.173  48221.848 -44143.911 144882.26
# 80353 80353  33730.895  24054.193 -13414.457  80876.25
# 80473 80473  32838.240  25585.930 -17309.261  82985.74
# 80483 80483  65278.330  50697.488 -34086.920 164643.58
# 83511 83511  19333.258  19333.258 -18559.231  57225.75
# 85233 85233   4158.448   4158.448  -3991.961  12308.86
# 8975   8975 138537.845 116014.341 -88846.085 365921.77
# 94259 94259  37468.505  37468.505 -35968.415 110905.43
# 94860 94860  36907.508  36907.508 -35429.879 109244.90
# 9517   9517  91381.129  48088.660  -2870.912 185633.17




# average cost 10 most frequent ecodes
EcodesAvgCost<-svyby(~adj_cost, ~ecode1, subset(svydes, ecode1 %in% rownames(ecodes)), svymean, na.rm
EcodesAvgCost
#        ecode1 adj_cost        se     ci_l      ci_u
```

```
#               592.2262 0.6967745 590.8605  593.5918
# E8120  E8120 997.8028 1.3035444 995.2479 1000.3577
# E8859  E8859 746.1664 0.5385309 745.1109  747.2219
# E8889  E8889 721.2775 0.6673637 719.9695  722.5855
# E9170  E9170 496.6702 0.6472324 495.4016  497.9387
# E9179  E9179 446.3013 0.3742304 445.5678  447.0348
# E9208  E9208 395.5892 0.3899572 394.8249  396.3535
# E927    E927 313.1359 0.2718108 312.6031  313.6686
# E9270  E9270 499.5250 0.5436503 498.4594  500.5905
# E9289  E9289 499.7797 0.5540140 498.6938  500.8655
gc()

# total cost top ten most frequent ecodes
system.time(EcodesTotCosts<-svyby(~adj_cost, ~ecode1, subset(svydes, ecode1 %in% rownames(ecodes), s
EcodesTotCosts
#      user    system   elapsed
# 25430.447  2152.401 14038.702
# > EcodesTotCosts
#       ecode1    adj_cost      se         ci_l        ci_u
#               5733086467 7740973  5717914439  5748258494
# E8120  E8120  6312303310 9891451  6292916422  6331690198
# E8859  E8859 10555086974 9587793 10536295245 10573878702
# E8889  E8889  7143848437 8183557  7127808960  7159887915
# E9170  E9170  1871123727 3200400  1864851059  1877396395
# E9179  E9179  4494660674 4818781  4485216036  4504105312
# E9208  E9208  2797996919 3563774  2791012050  2804981789
# E927    E927  1841596025 2258916  1837168630  1846023419
# E9270  E9270  2085055100 3158735  2078864093  2091246108
# E9289  E9289  3932524454 5282273  3922171390  3942877518
gc()

# average cost most expensive ecodes
system.time(avgExpensEcodes<-svyby(~adj_cost, ~ecode1, subset(svydes, ecode1 %in% rownames(expensive
avgExpensEcodes
#      user    system   elapsed
# 13515.744  1030.751   594.849
# > avgExpensEcodes
#       ecode1 adj_cost        se        ci_l     ci_u
# E8030  E8030 1980.551 1544.0477 -1045.72667 5006.829
# E8302  E8302 2603.188 1094.3509   458.29919 4748.076
# E8336  E8336 1664.752 1018.0891  -330.66614 3660.170
# E8372  E8372 3608.544 2371.3263 -1039.17017 8256.258
# E8402  E8402 2076.151 1032.3175    52.84622 4099.456
# E8760  E8760 1560.916  503.8415   573.40442 2548.427
# E9396  E9396 1424.336  571.7942   303.64030 2545.032
# E9511  E9511 1639.923  252.9494  1144.15144 2135.695
# E9518  E9518 2161.443 1038.8276   125.37795 4197.507
```

```
# E9920  E9920 3764.672    0.0000  3764.67200 3764.672
# E9922  E9922 2001.647    0.0000  2001.64683 2001.647
# E9929  E9929 4226.527 2033.4275   241.08211 8211.972
>
gc()

# total cost most expensive ecodes
system.time(totExpensEcodes<-svyby(~adj_cost, ~ecode1, subset(svydes, ecode1 %in% rownames(expensive
totExpensEcodes
#     user   system  elapsed
# 1007.916  114.005  595.382
# > totExpensEcodes
#       ecode1   adj_cost         se       ci_l      ci_u
# E8030  E8030  45990.957  41890.152 -36112.233 128094.15
# E8302  E8302  95459.071  53466.214  -9332.783 200250.92
# E8336  E8336  13957.340  12088.489  -9735.663  37650.34
# E8372  E8372 154079.537 123655.567 -88280.921 396439.99
# E8402  E8402  76222.884  45992.710 -13921.172 166366.94
# E8760  E8760  72538.470  32459.363   8919.287 136157.65
# E9396  E9396  25091.280  15490.400  -5269.346  55451.91
# E9511  E9511  15229.022  10964.349  -6260.707  36718.75
# E9518  E9518  22352.267  18314.149 -13542.805  58247.34
# E9920  E9920  21242.514  21242.514 -20392.048  62877.08
# E9922  E9922   7743.608   7743.608  -7433.585  22920.80
# E9929  E9929  36281.611  32677.060 -27764.250 100327.47
gc()
```

| | Year | Age Group | Count | SE |
|---|---|---|---|---|
| 1 | 2006 | Younger than 18 | 7457903.00 | 5963.93 |
| 2 | 2007 | | 7379373.00 | 5938.79 |
| 3 | 2008 | | 7072187.00 | 5649.37 |
| 4 | 2009 | | 6920894.00 | 5756.46 |
| 5 | 2010 | | 6761431.00 | 5637.68 |
| 6 | 2011 | | 6845915.00 | 5466.50 |
| 7 | 2012 | | 6951220.00 | 5561.03 |
| 8 | 2006 | 18 to 45 | 11297852.00 | 7218.43 |
| 9 | 2007 | | 11117346.00 | 7179.34 |
| 10 | 2008 | | 10860471.00 | 6877.82 |
| 11 | 2009 | | 10294296.00 | 6965.59 |
| 12 | 2010 | | 10531512.00 | 6952.26 |
| 13 | 2011 | | 10072285.00 | 6444.13 |
| 14 | 2012 | | 10024540.00 | 6604.61 |
| 15 | 2006 | 45 to 55 | 4745698.00 | 4830.58 |
| 16 | 2007 | | 4826772.00 | 4872.49 |
| 17 | 2008 | | 4865721.00 | 4742.89 |
| 18 | 2009 | | 4808729.00 | 4881.30 |
| 19 | 2010 | | 5043254.00 | 4963.50 |
| 20 | 2011 | | 4957956.00 | 4710.57 |
| 21 | 2012 | | 4962435.00 | 4763.83 |
| 22 | 2006 | 65 to 85 | 2318552.00 | 3401.10 |
| 23 | 2007 | | 2357144.00 | 3436.13 |
| 24 | 2008 | | 2400043.00 | 3373.40 |
| 25 | 2009 | | 2435519.00 | 3520.23 |
| 26 | 2010 | | 2485301.00 | 3535.61 |
| 27 | 2011 | | 2501217.00 | 3394.97 |
| 28 | 2012 | | 2555035.00 | 3454.13 |
| 29 | 2006 | Older than 85 | 804011.80 | 2013.52 |
| 30 | 2007 | | 838886.80 | 2072.28 |
| 31 | 2008 | | 886432.50 | 2064.62 |
| 32 | 2009 | | 916916.00 | 2179.21 |
| 33 | 2010 | | 950394.80 | 2200.10 |
| 34 | 2011 | | 969965.80 | 2143.67 |
| 35 | 2012 | | 977223.10 | 2149.26 |

Table 2: Emergency Department Visits for Traumatic Injury by Age Group and Year, US Hospitals, 2006-2012.

| Variable | Odds Ratio (95% CI) |
|---|---|
| Age Group | 0.9952900 ( 0.963610 , 1.028000) |
| Female | 0.9324100 ( 0.922560 , 0.942370) |
| Teaching Hospital | 1.0695000 ( 0.804190 , 1.422400) |
| Trauma Center | 0.9655500 ( 0.789000 , 1.181600) |
| Severe Injury | 4.1926000 ( 3.873300 , 4.538200) |
| Charlson Score | 10.3470000 ( 7.119200 , 15.037000) |

Table 3: Survey-Adjusted Logistic Regression Results. Factors Associated with Injury Mortality. Emergency Department Visits for Traumatic Injury, US Hospitals, 2006-2012.

| Injury Category | Count (s.e.) |
|---|---|
| Superficial Contusions | 39055035.87 (12353.6236) |
| Sprains / Strains | 43306062.31 (12762.9627) |
| Open Wounds | 40534722.17 (12534.9346) |
| System Wide | 4303484.41 ( 4581.3600) |
| Unspecified | 12687322.21 ( 7591.1916) |
| Fractures | 28836088.28 (10999.5770) |
| Dislocation | 3247192.74 ( 4004.4218) |
| Amputations | 369822.98 ( 1371.0250) |
| Internal Organ | 5155054.02 ( 5022.9648) |
| Burns | 2881833.83 ( 3788.5960) |
| Crushing | 667234.87 ( 1852.3627) |
| Blood Vessels | 69722.47 ( 597.3702) |
| Nerves | 80855.26 ( 638.5664) |

Table 4: Barell Matrix Categories Emergency Department Visits for Traumatic Injury, US Hospitals, 2006-2012.

| Injury Cause | Count (s.e.) |
|---|---|
| Cutting / Piercing | 14717767.68 ( 8209.7518) |
| Drowning / Submersion | 32582.80 ( 410.1982) |
| Fire / Burn | 2478927.95 ( 3517.1305) |
| Machinery | 918969.45 ( 2147.3494) |
| Pedal Cyclist | 2254208.56 ( 3348.3867) |
| Pedestrian | 141375.84 ( 838.5714) |
| Non Motor-Vehicle Transport | 2296248.10 ( 3416.8735) |
| Natural Environment | 3248078.71 ( 3988.7310) |
| Overexertion | 15686930.95 ( 8410.4226) |
| Struck By / Against | 26876289.60 (10650.0292) |
| Suffocation | 222728.79 ( 1058.7297) |

Table 5: Injury Causes Classified by Agency for Healthcare Research and Quality Clinical Classification Software. Emergency Department Visits for Traumatic Injury, US Hospitals, 2006-2012.

| Diagnosis | Average Cost (95% CI) | Total Cost (95% CI) |
|---|---|---|
| Knee or Leg Sprain | 436.8201 (435.8858, 437.7543) | 1498530231 (1493838041, 1503222422) |
| Ankle Sprain | 433.4812 (432.8830, 434.0793) | 2827514797 (2821466650, 2833562944) |
| Neck Sprain | 801.4461 (799.7869, 803.1053) | 5838335298 (5823220988, 5853449607) |
| Back Sprain | 465.6632 (464.4153, 466.9111) | 2364346968 (2356590262, 2372103674) |
| Open Scalp Wound | 786.3268 (783.6657, 788.9879) | 2557131866 (2546573874, 2567689859) |
| Open Forehead Wound | 675.4546 (673.3406, 677.5686) | 2649792857 (2639746915, 2659838798) |
| Open Hand Wound | 426.5597 (425.1612, 427.9582) | 1306872991 (1301544122, 1312201859) |
| Open Finger Wound | 372.5561 (371.9571, 373.1550) | 2627524210 (2621599972, 2633448449) |
| Face/Head/Neck Contusion | 852.0335 (850.2552, 853.8117) | 5376155444 (5361819961, 5390490926) |
| Head Injury | 1107.9382 1(105.6565, 1110.2199) | 6066082336 (6049541829, 6082622843) |

Table 6: Average and Total Cost Ten Most Frequent Diagnoses. Emergency Department Visits for Traumatic Injury, US Hospitals, 2006-2012.

| Diagnosis | Average Cost (95% CI) | Total Cost (95% CI) |
|---|---|---|
| Closed Skull Fracture - Coma | 5332.114 ( 450.8328, 10213.395) | 50369.173 (0, 144882.26) |
| Open Skull Fracture - Coma | 4002.267 ( 2807.5407, 5196.993) | 33730.895 (0, 80876.25) |
| Open Skull Fracture / Other - Coma | 3850.481 ( 896.0716, 6804.891) | 32838.240 (0, 82985.74) |
| Open Skull Fracture / Other | 3956.723 ( 0 , 8394.652) | 65278.330 (0, 164643.58) |
| Open Posterior Hip Dislocation | 4430.500 ( 4430.5001 , 4430.500) | 19333.258 (0, 57225.75) |
| Open Subdural Hematoma - Coma | 4119.218 ( 4119.2180 , 4119.218) | 4158.448 ( 0, 12308.86) |
| Leg Amputation | 6493.240 (-2438.0762, 15424.557) | 138537.845 (0, 365921.77) |
| Third Degree Burn of Trunk | 7602.000 ( 7602.0000 , 7602.000) | 37468.505 (0, 110905.43) |
| Third Degree Burn GT 70% | 5359.609 ( 5359.6094 , 5359.609) | 36907.508 (0, 109244.90) |
| Hypoglossal Nerve Injury | 3558.539 ( 797.5001 , 6319.578) | 91381.129 ( 0, 185633.17) |

Table 7: Average and Total Cost Ten Most Expensive Diagnoses. Emergency Department Visits for Traumatic Injury, US Hospitals, 2006-2012.