CITS4009 - Project 1

Code

Wenxiao Zhang (22792191)

Introduction

The data set analyzed is US Accident Injury Dataset, obtained from LMS. The entire dataset spans across 15 years (2000 to 2015), and has a total of 202,814 observations.

Data loading, overview and set up

Load libraries

Hide

```
library(ggplot2)
library(gridExtra)
library(dplyr)
library(ggthemes)
library(vtreat)
library(chron)
library(crayon)
library(lubridate)
```

Setting up a plotting theme so that all charts look coherent

Hide

```
my_theme <- theme_few(base_size = 12)+
  theme(plot.title = element_text(color = "#000080")) +
  theme(plot.margin=margin(10,30,10,30))
```

Load the main data

Hide

```
us_data <-read.csv('./data/us_data.csv')
```

Hide

```
head(us_data)
```

	MINE… <int>	CONTROLLE… <chr>	CONTROLLER_… <chr>	OPERATO… <chr>	OPERATOR_NAME <chr>
1	100003	41044	Lhoist Group	L13586	Lhoist North America
2	100003	41044	Lhoist Group	L13586	Lhoist North America
3	100008	M31753	Alan B Cheney	L31753	Cheney Lime & Cement Company
4	100011	M11763	Imerys S A	L17074	Imerys Pigments LLC
5	100011	M11763	Imerys S A	L17074	Imerys Pigments LLC
6	100011	M11763	Imerys S A	L17074	Imerys Pigments LLC

6 rows | 1-7 of 57 columns

Using str to analyze the data

Hide

```
str(us_data)
```

There are 202,814 observations with 57 variables, comprised by 39 factors, 13 integers, and 5 numeric variables. In terms of factors, apart from IDs and CDs which could be ignored for now, many of them have too many values or levels. In addition, some of them have a considerable number of missing values. We will deal with these later on. As for numeric variables, ACCIDENT_TIME and schedule charge are variables with special values that need to be dealt with.

Using summary to analyze the data

Hide

```
summary(us_data)
```

Like TOT_EXPER DAYS_RESTRICT, and DAYS_LOST, most of the numeric variables have many NA value, and they tend to be right-skewed.

Viewing the first six observations

Hide

```
head(us_data)
```

	MINE… <int>	CONTROLLE… <chr>	CONTROLLER_… <chr>	OPERATO… <chr>	OPERATOR_NAME <chr>
1	100003	41044	Lhoist Group	L13586	Lhoist North America
2	100003	41044	Lhoist Group	L13586	Lhoist North America
3	100008	M31753	Alan B Cheney	L31753	Cheney Lime & Cement Company
4	100011	M11763	Imerys S A	L17074	Imerys Pigments LLC
5	100011	M11763	Imerys S A	L17074	Imerys Pigments LLC
6	100011	M11763	Imerys S A	L17074	Imerys Pigments LLC

6 rows | 1-7 of 57 columns

These two summaries show that NARRATIVE is a description of the accident/injury/illness, which means it is a free text field.

Initial transformations

Based on the above observations, we'll convert some columns to a more appropriate format and add new ones that might be useful in further analysis

Hide

```
# apply vtreat to deal with missing values
treatment_plan <- design_missingness_treatment(us_data, varlist = c("DAYS_LOST", "DAYS_RESTRI
CT", "NO_INJURIES"))
us_data <- prepare(treatment_plan, us_data)

us_data <- within(us_data, {

  ACCIDENT_DT <- as.Date(ACCIDENT_DT, format="%d/%m/%Y")

  ACCIDENT_TIME <- times(sprintf( "%d:%02d:00", ACCIDENT_TIME %/% 100, ACCIDENT_TIME %% 100
 ))

  SCHEDULE_CHARGE.fix <- as.factor(ifelse(is.na(SCHEDULE_CHARGE), "MISSING VALUE",
                                          ifelse(SCHEDULE_CHARGE==0, SCHEDULE_CHARGE, "OTHER
 VALUE")))
  UG_LOCATION.fix <-
    ifelse(UG_LOCATION=="NO VALUE FOUND" | UG_LOCATION=="NOT MARKED" | UG_LOCATION=="OTHER",
 UG_LOCATION, "VALID VALUE")

  UG_MINING_METHOD.fix <-
    ifelse(UG_MINING_METHOD=="NO VALUE FOUND", UG_MINING_METHOD,"VALID VALUE")

  IMMED_NOTIFY.fix <-
    ifelse(IMMED_NOTIFY=="NO VALUE FOUND" | IMMED_NOTIFY=="NOT MARKED", IMMED_NOTIFY,"VALID V
ALUE")

  CAL_QTR<-factor(CAL_QTR)

  CAL_YR<-factor(CAL_YR)
})



# a function to set the column name after doing transformation
set_col_name<-function(df, month=0){
  if(month==0){
    colnames(df) <- c("ACCIDENT_YEAR", colnames(df)[2:length(colnames(df))])
  }else{
    colnames(df) <- c("ACCIDENT_YEAR", "ACCIDENT_MONTH", colnames(df)[3:length(colnames(d
f))])
  }
  return(df)
}


# a function to convert invalid value to NA and remove NA in a dataframe
set_remove_NA<-function(df,col_name, opt=0){
  for(col in col_name){
    for(row in 1:NROW(df[,col])){
      if(df[row,col] == ""||df[row,col]=="NO VALUE FOUND"){
        df[row,col]=NA
      }
      else if(opt == 1 & df[row,col] == "NOT MARKED"){
        df[row,col]=NA
```

```
      }
    }
  }
  df<-na.omit(df)
  return(df)
}
```

Analyze the number of NA each variables

Hide

```
apply(is.na(us_data), 2, sum)
```

```
             MINE_ID        CONTROLLER_ID      CONTROLLER_NAME           OPERATOR_ID        OP
ERATOR_NAME        CONTRACTOR_ID
                   0                    0                    0                    0
0                    0
         DOCUMENT_NO           SUBUNIT_CD              SUBUNIT          ACCIDENT_DT
CAL_YR              CAL_QTR
                   0                    0                    0                    0
0                    0
           FISCAL_YR           FISCAL_QTR        ACCIDENT_TIME     DEGREE_INJURY_CD        DE
GREE_INJURY        FIPS_STATE_CD
                   0                    0                16569                    0
0                    0
      UG_LOCATION_CD          UG_LOCATION  UG_MINING_METHOD_CD     UG_MINING_METHOD      MINI
NG_EQUIP_CD         MINING_EQUIP
                   0                    0                    0                    0
0                    0
        EQUIP_MFR_CD       EQUIP_MFR_NAME       EQUIP_MODEL_NO     SHIFT_BEGIN_TIME    CLASSI
FICATION_CD       CLASSIFICATION
                   0                    0                   48                  991
0                    0
    ACCIDENT_TYPE_CD        ACCIDENT_TYPE            TOT_EXPER           MINE_EXPER
JOB_EXPER        OCCUPATION_CD
                   0                    0                37400                34325
33746                    0
          OCCUPATION          ACTIVITY_CD             ACTIVITY     INJURY_SOURCE_CD        IN
JURY_SOURCE     NATURE_INJURY_CD
                   0                    0                    0                    0
0                    0
       NATURE_INJURY     INJ_BODY_PART_CD        INJ_BODY_PART      SCHEDULE_CHARGE
TRANS_TERM    RETURN_TO_WORK_DT
                   0                    0                    0                65006
0                    0
     IMMED_NOTIFY_CD         IMMED_NOTIFY      INVEST_BEGIN_DT            NARRATIVE        CL
OSED_DOC_NO       COAL_METAL_IND
                   0                    0                    0                    0
116621                    0
           DAYS_LOST      DAYS_LOST_isBAD        DAYS_RESTRICT  DAYS_RESTRICT_isBAD
NO_INJURIES     IMMED_NOTIFY.fix
                   0                    0                    0                    0
0                    0
UG_MINING_METHOD.fix      UG_LOCATION.fix  SCHEDULE_CHARGE.fix
                   0                    0                    0
```

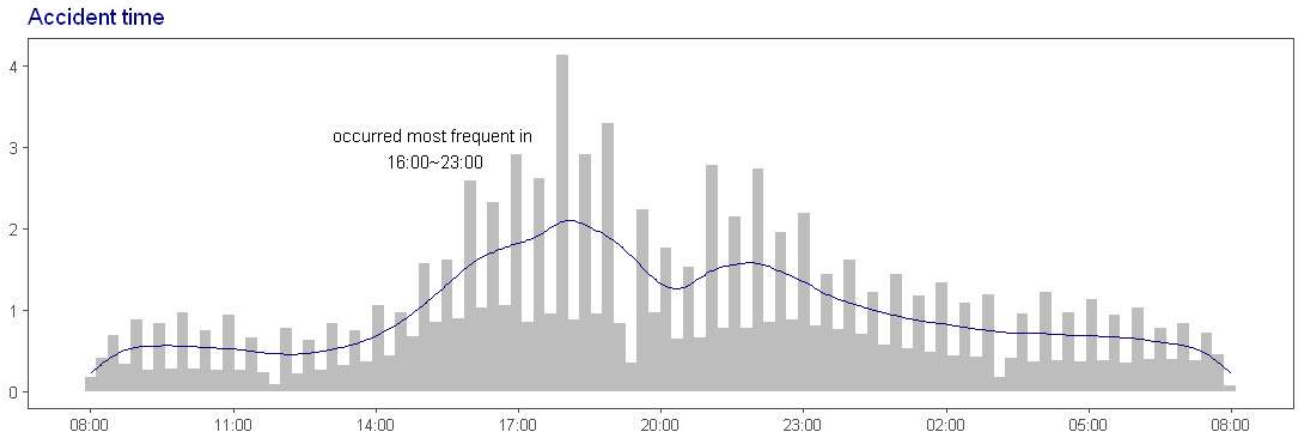
Most of the columns don’t have any NA. The ones that do might be due to loss of accident information.

Analyzing accident time and accident date

We’ll first analyze the distribution of the accident amount with time period and date to see when did accidents occur frequently. We’ll use historgram to analyse this. As historgram can clearly show the distribution of a single continous variable.

Hide

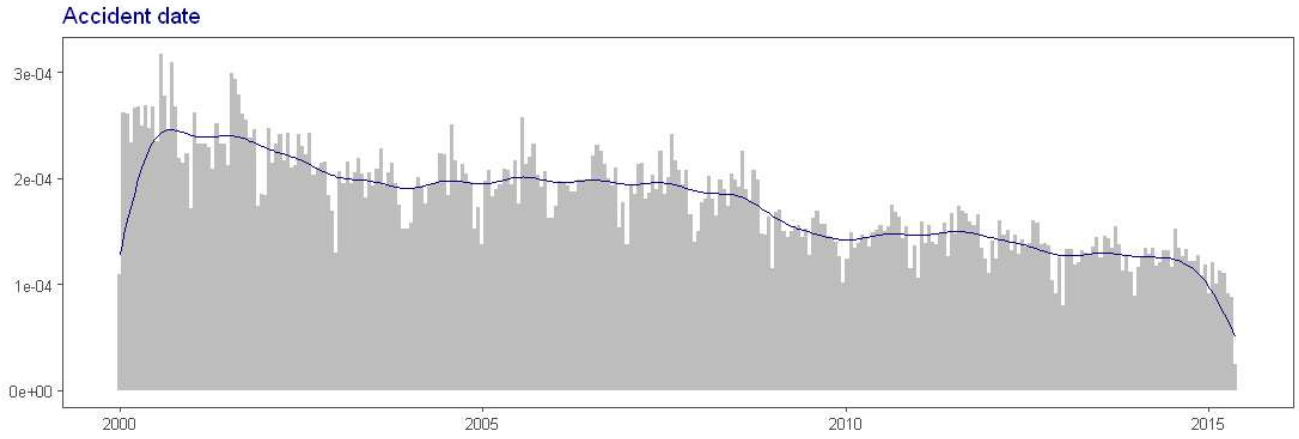
```
ggplot(us_data, aes(x=ACCIDENT_TIME)) +
  geom_histogram(aes(y = ..density..),bins=100, fill = "grey")+
  geom_density(color="#000080")+
  scale_x_chron(format="%H:%M",n=9)+
   labs(x = NULL, y = NULL) +
  annotate("text", x = 0.3, y = 3, label = "occurred most frequent in\n 16:00~23:00") +
  ggtitle("Accident time")+
  my_theme
```



Accidents occurred mainly in 16:00 ~ 23:00.

Hide

```
ggplot(us_data, aes(x=ACCIDENT_DT)) +
  geom_histogram(aes(y = ..density..),bins=300, fill = "grey")+
  geom_density(color="#000080")+
   labs(x = NULL, y = NULL) +
  ggtitle("Accident date")+
  my_theme
```

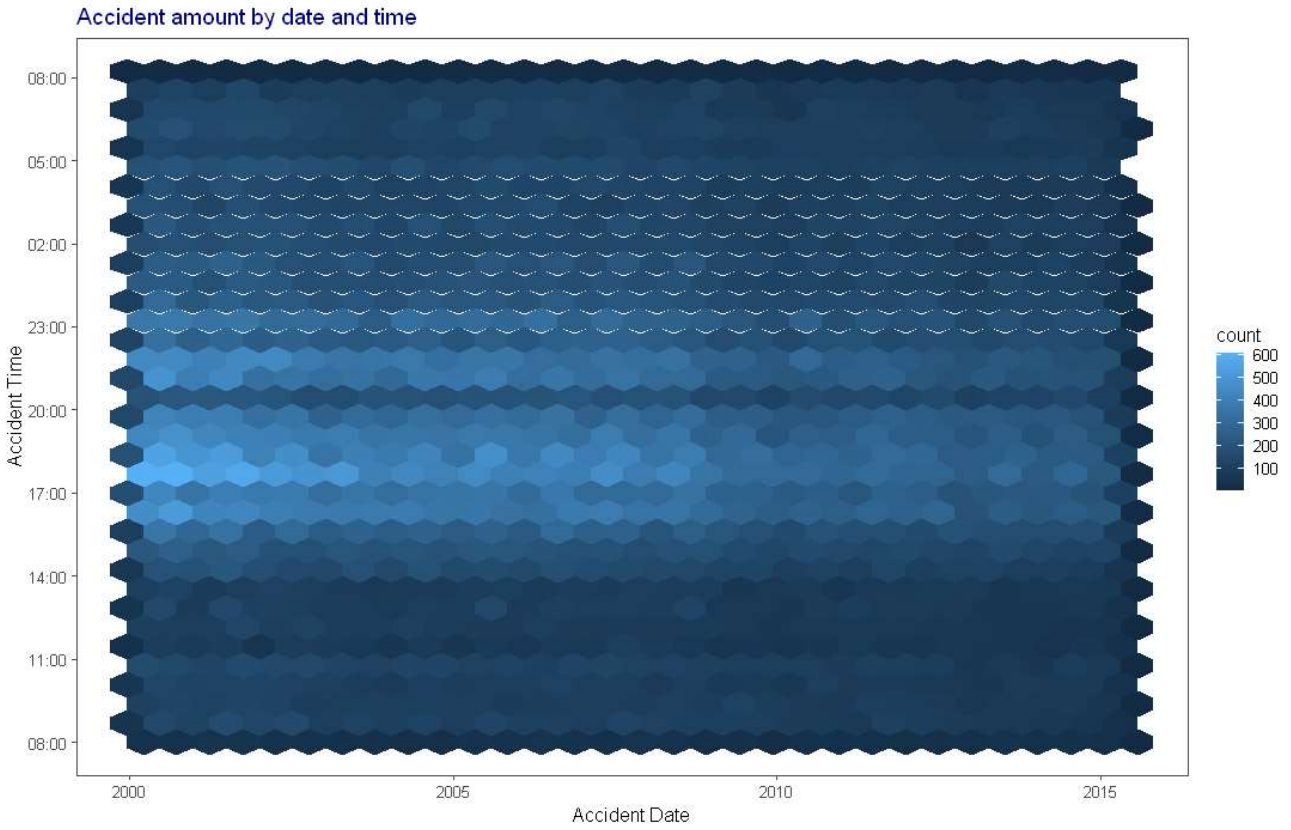


The distribution of accident amount shows a decreasing trend by date.

Now we'll use hex plot to visualize the distribution of the amount of accident by date and time. We'll use geom_hex to analyse this, as it can show the density between two continuous variables.

Hide

```
ggplot(us_data, aes(x=ACCIDENT_DT, y=ACCIDENT_TIME)) +
  geom_hex(na.rm = T, bins=30)+
  scale_y_chron(format="%H:%M",n=9)+
   labs(x = "Accident Date", y = "Accident Time") +
  ggtitle("Accident amount by date and time")+
  my_theme
```



We can see that the accident amount distributed intensively in left-middle region of the graph, which is around 16:00~23:00 and 2000~2005.

Analyzing injuries data

As the values of NO_INJURIES variable were mostly 0 and 1, we'll analyze it by month.

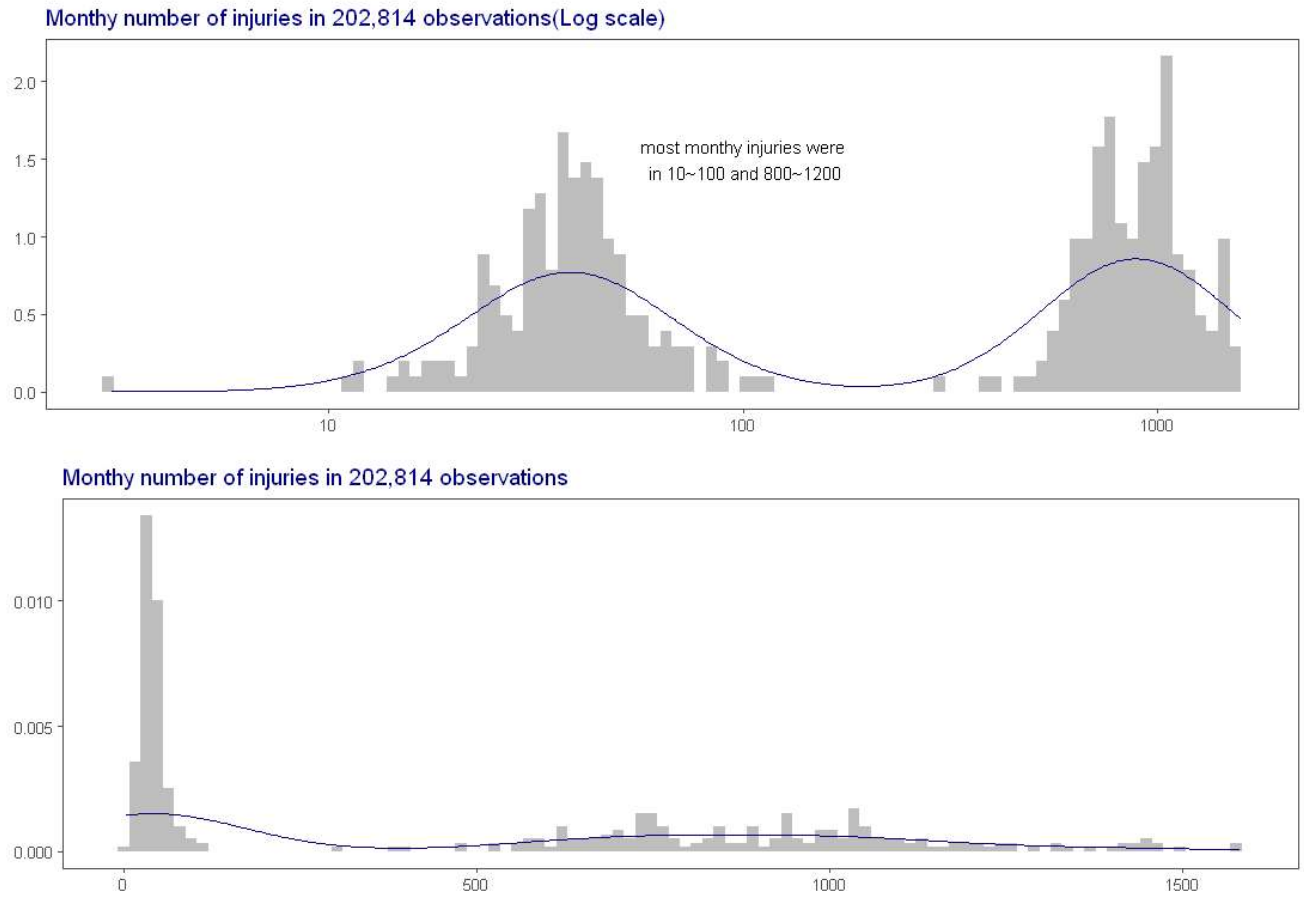
Hide

```
#initially transform data
us_data_by_injury <- us_data[, c('ACCIDENT_DT','NO_INJURIES')] %>%
  group_by(year(ACCIDENT_DT), month(ACCIDENT_DT)) %>%
  summarise(NO_INJURIES_SUM = sum(NO_INJURIES))
us_data_by_injury<-set_col_name(us_data_by_injury_result, 1)
```

Firstly, we'll use histogram to analyse it.

Hide

```
p1<-ggplot(us_data_by_injury, aes(x=NO_INJURIES_SUM)) +
  geom_histogram(aes(y = ..density..),bins=100, fill = "grey")+
  geom_density(color="#000080")+
  scale_x_continuous(trans='log10')+
   labs(x = NULL, y = NULL) +
  annotate("text", x = 100, y = 1.5, label = "most monthy injuries were\n in 10~100 and 800~1
200") +
  ggtitle("Monthy number of injuries in 202,814 observations(Log scale)")+
  my_theme

p2<-ggplot(us_data_by_injury, aes(x=NO_INJURIES_SUM)) +
  geom_histogram(aes(y = ..density..),bins=100, fill = "grey")+
  geom_density(color="#000080")+
  labs(x = NULL, y = NULL) +
  ggtitle("Monthy number of injuries in 202,814 observations")+
  my_theme



grid.arrange(p1,p2, ncol=1)
```

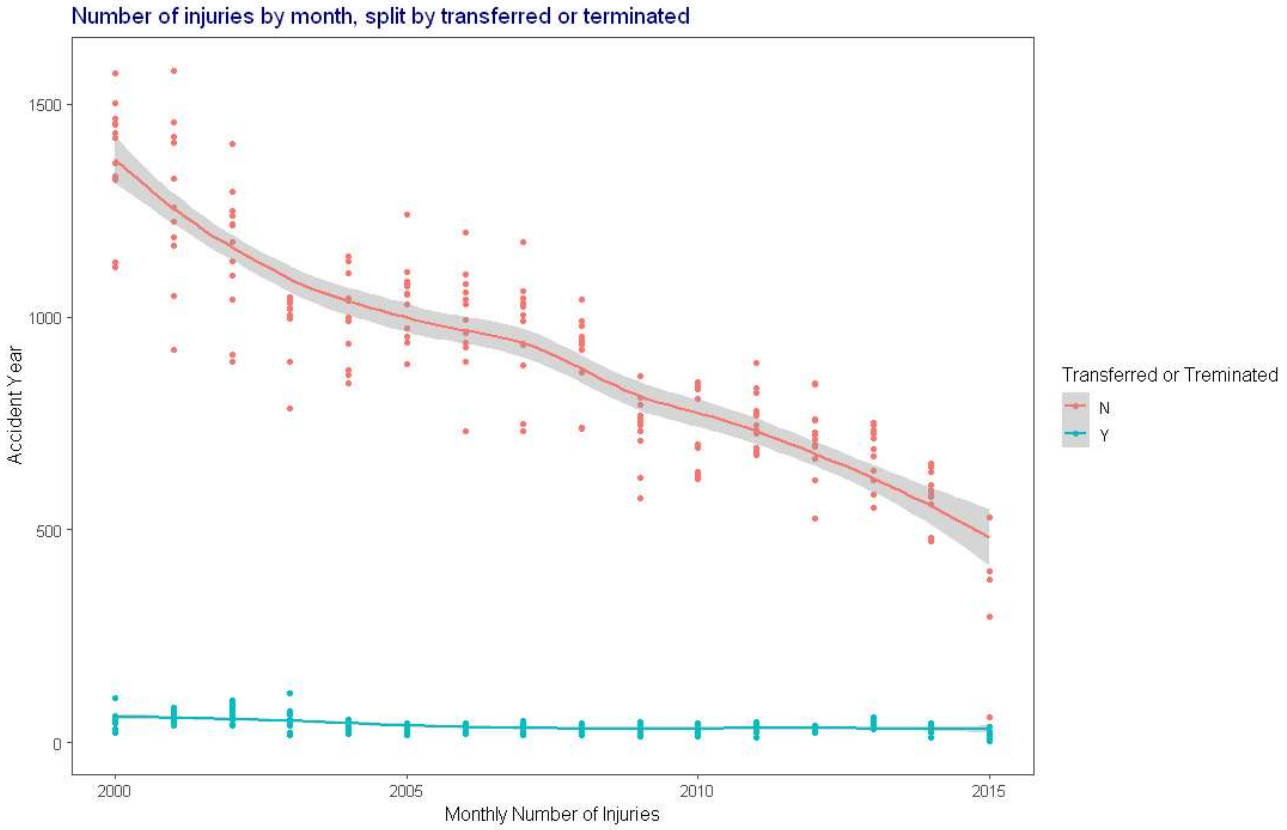


The monthly number of injuries is highly right-skewed.

Now we'll look at the number of injuries by month split by different underground mining method to see which method caused more injuries. We'll use scatter and smooth plot to visualize the continuous variable to show the distribution and general trend.

Hide

```
#initially transform data
us_data_by_injury_result <- us_data[, c('ACCIDENT_DT','NO_INJURIES','TRANS_TERM')] %>%
  group_by(year(ACCIDENT_DT), month(ACCIDENT_DT), TRANS_TERM) %>%
  summarise(NO_INJURIES_SUM = sum(NO_INJURIES))
us_data_by_injury_result<-set_col_name(us_data_by_injury_result, 1)

# convert meaningless value(NO VALUE FOUND) to NA and then delete them
us_data_by_injury_result<-set_remove_NA(us_data_by_injury_result, "TRANS_TERM")

ggplot(us_data_by_injury_result, aes(x=ACCIDENT_YEAR, y=NO_INJURIES_SUM, color=TRANS_TERM, gr
oup=TRANS_TERM)) +
  geom_point()+
  geom_smooth()+
  labs(x = "Monthly Number of Injuries", y = "Accident Year", color="Transferred or Treminate
d") +
  ggtitle("Number of injuries by month, split by transferred or terminated")+
  my_theme
```

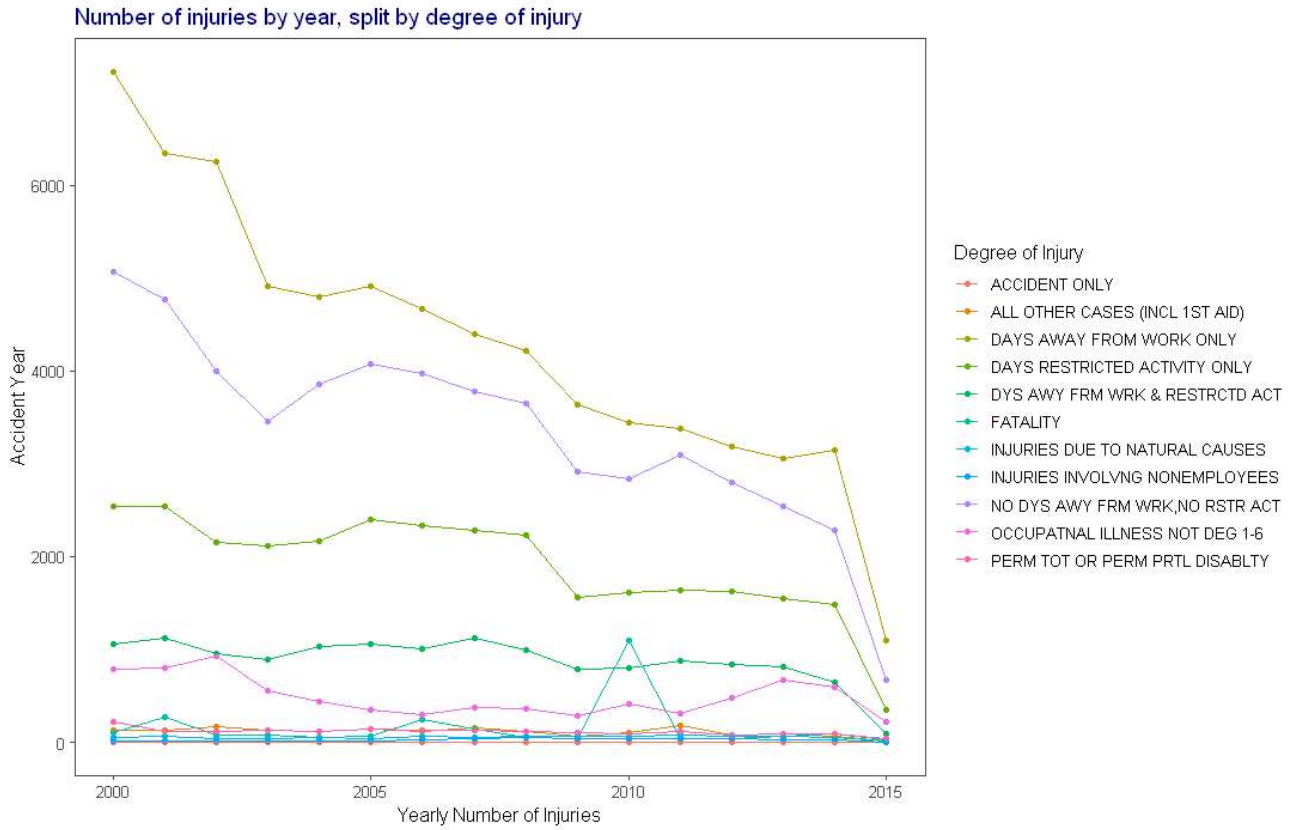


most injuries/illnesses were not being permanently transferred or terminated, and the total number of injury is decreasing by date.

Now we'll look at the number of injuries by year split by different degrees of injury to see which degree of injury happened frequently. We'll use line and point plot to show the trend between each year.

Hide

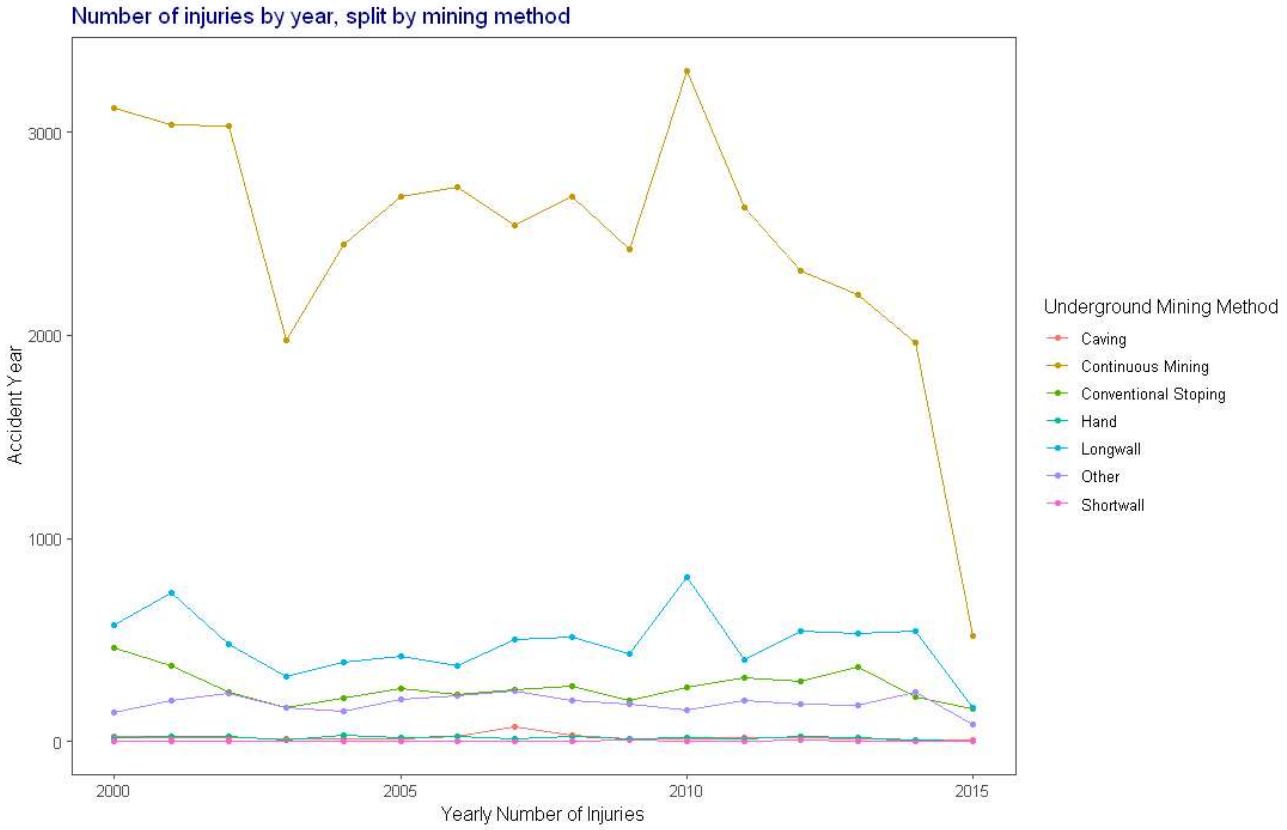
```
#initially transform data
us_data_by_injury_degree <- us_data[, c('ACCIDENT_DT','DEGREE_INJURY','NO_INJURIES')] %>%
  group_by(year(ACCIDENT_DT), DEGREE_INJURY) %>%
  summarise(NO_INJURIES_SUM = sum(NO_INJURIES))
us_data_by_injury_degree <-set_col_name(us_data_by_injury_degree)
# convert meaningless value(NO VALUE FOUND) to NA and then delete them
us_data_by_injury_degree<-set_remove_NA(us_data_by_injury_degree,"DEGREE_INJURY")

ggplot(us_data_by_injury_degree, aes(x=ACCIDENT_YEAR, y=NO_INJURIES_SUM, color=DEGREE_INJURY,
group=DEGREE_INJURY)) +
  geom_point()+
  geom_line()+
   labs(x = "Yearly Number of Injuries", y = "Accident Year", color="Degree of Injury") +
  ggtitle("Number of injuries by year, split by degree of injury")+
  my_theme
```



Then let's look at the number of injuries by year split by different underground mining method to see which method caused more injuries.

Hide

```
#initially transform data
us_data_by_injury_method <- us_data[, c('ACCIDENT_DT','UG_MINING_METHOD','NO_INJURIES')] %>%
  group_by(year(ACCIDENT_DT), UG_MINING_METHOD) %>%
  summarise(NO_INJURIES_SUM = sum(NO_INJURIES))
us_data_by_injury_method <- set_col_name(us_data_by_injury_method)

# convert meaningless value(NO VALUE FOUND) to NA and then delete them
us_data_by_injury_method<-set_remove_NA(us_data_by_injury_method,"UG_MINING_METHOD")


ggplot(us_data_by_injury_method, aes(x=ACCIDENT_YEAR, y=NO_INJURIES_SUM, color=UG_MINING_METH
OD, group=UG_MINING_METHOD)) +
  geom_point()+
  geom_line()+
  labs(x = "Yearly Number of Injuries", y = "Accident Year", color="Underground Mining Metho
d") +
  ggtitle("Number of injuries by year, split by mining method")+
  my_theme
```

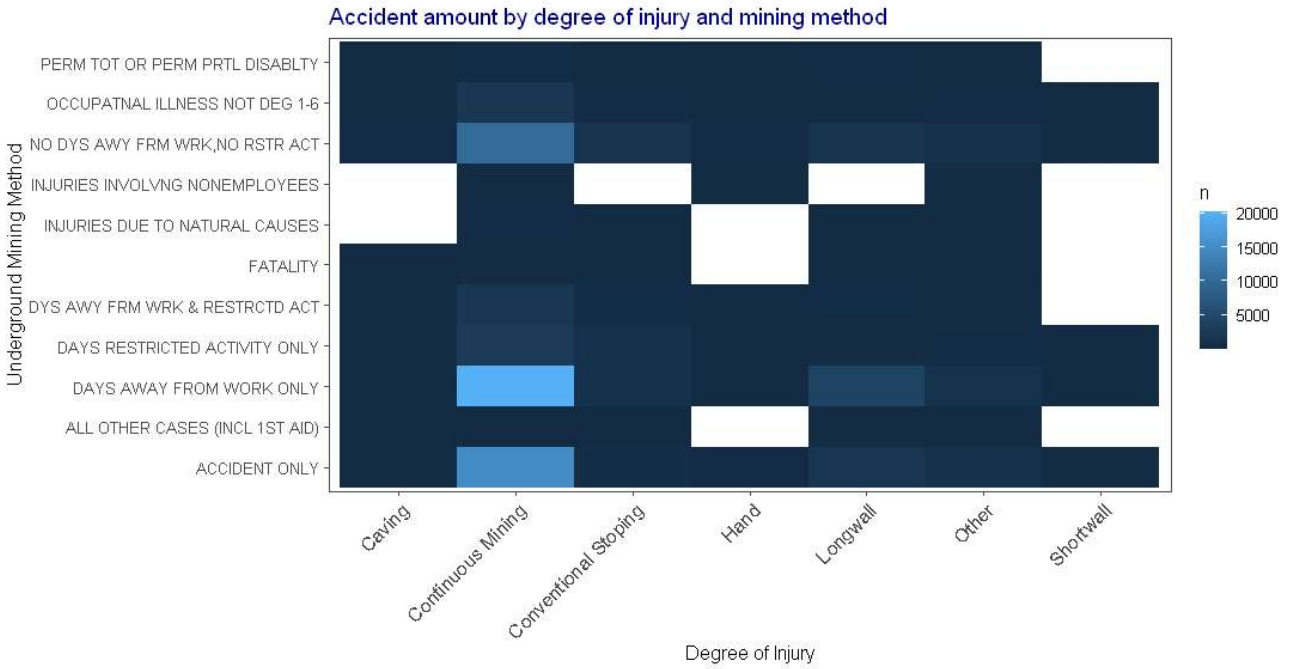


From the three graphs above we can see that the monthly injuries were relatively high only in some special values of factors, the rest were basically 0 during the period, and they all show a decreasing trend over these years.

use degree of injury and underground mining method to visualize the distribution of accident amount We'll use geom_tile to analyze this. As geom_tile can clearly show the relationship between two categorical variables.

Hide

```
injury_by_method_count <- count(us_data, DEGREE_INJURY, UG_MINING_METHOD)

injury_by_method_count<-set_remove_NA(injury_by_method_count, c("DEGREE_INJURY","UG_MINING_METHOD"))

ggplot(injury_by_method_count, aes(x=UG_MINING_METHOD, y=DEGREE_INJURY))+
  geom_tile(aes(fill=n))+
  labs(x = "Degree of Injury", y = "Underground Mining Method") +
  ggtitle("Accident amount by degree of injury and mining method")+
  my_theme+
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 12))
```

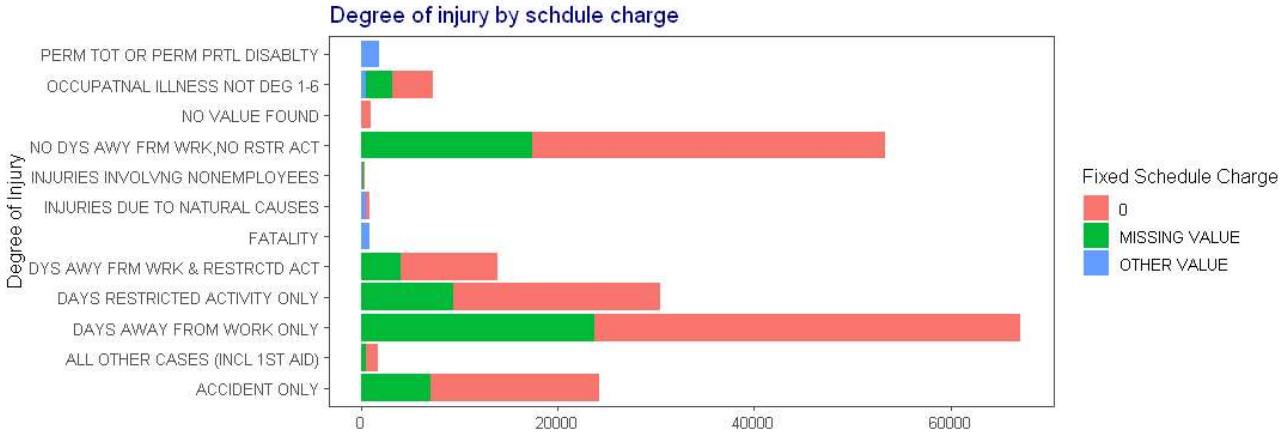


As expected, those values caused higher number of injuries have more accident amount than others. Such as 'DAYS AWAY FROM WORK ONLY' by 'Continuous Mining'.

Use degree of injury and schedule charge to visualize the distribution apply stacked bar plot to deal with this can clearly show the amount of each type of schedule charge.

Hide

```
ggplot(us_data, aes(y=DEGREE_INJURY))+
  geom_bar(aes(fill=SCHEDULE_CHARGE.fix))+
   labs(x = NULL, y = "Degree of Injury", fill="Fixed Schedule Charge") +
  ggtitle("Degree of injury by schdule charge")+
  my_theme
```



we can see most values of schedule charge based on most levels of DEGREE_INJURY are 0 and MISSING VALUE. This means that there is no permanent injury/illness in most case scenario. SCHEDULE_CHARGE may only be applied when some special degrees of injury occurred, such as 'PERM TOT OR PERM PRTL DISABLTY' and 'FATALITY'.

Analyzing days lost data

As the original DAYS_LOST and DAYS_RESTRICT contains too much NAs and 0 values, we'll analyze them by grouping them with year and month and then sum them up in order to have graphs which can show information clearer.

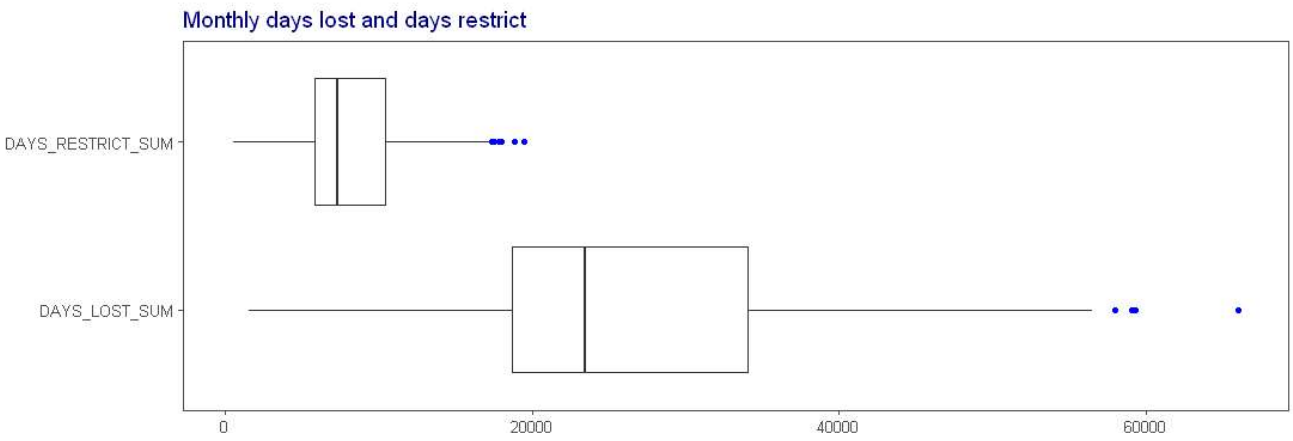
Hide

```
#initially transform data
us_data_by_days <- us_data[, c('ACCIDENT_DT','DAYS_LOST', 'DAYS_RESTRICT')] %>%
  group_by(year(ACCIDENT_DT), month(ACCIDENT_DT)) %>%
  summarise(DAYS_LOST_SUM = sum(DAYS_LOST),DAYS_RESTRICT_SUM = sum(DAYS_RESTRICT))
  us_data_by_days <- set_col_name(us_data_by_days)
```

Firstly, we'll use boxplot and histogram to analyze monthly days lost and days restrict data

Hide

```
ggplot(stack(us_data_by_days[,3:4])) +
  geom_boxplot(aes(x=ind, y=values), outlier.color = "blue")+
   labs(x = NULL, y = NULL) +
  ggtitle("Monthly days lost and days restrict") +
    coord_flip()+
  my_theme
```

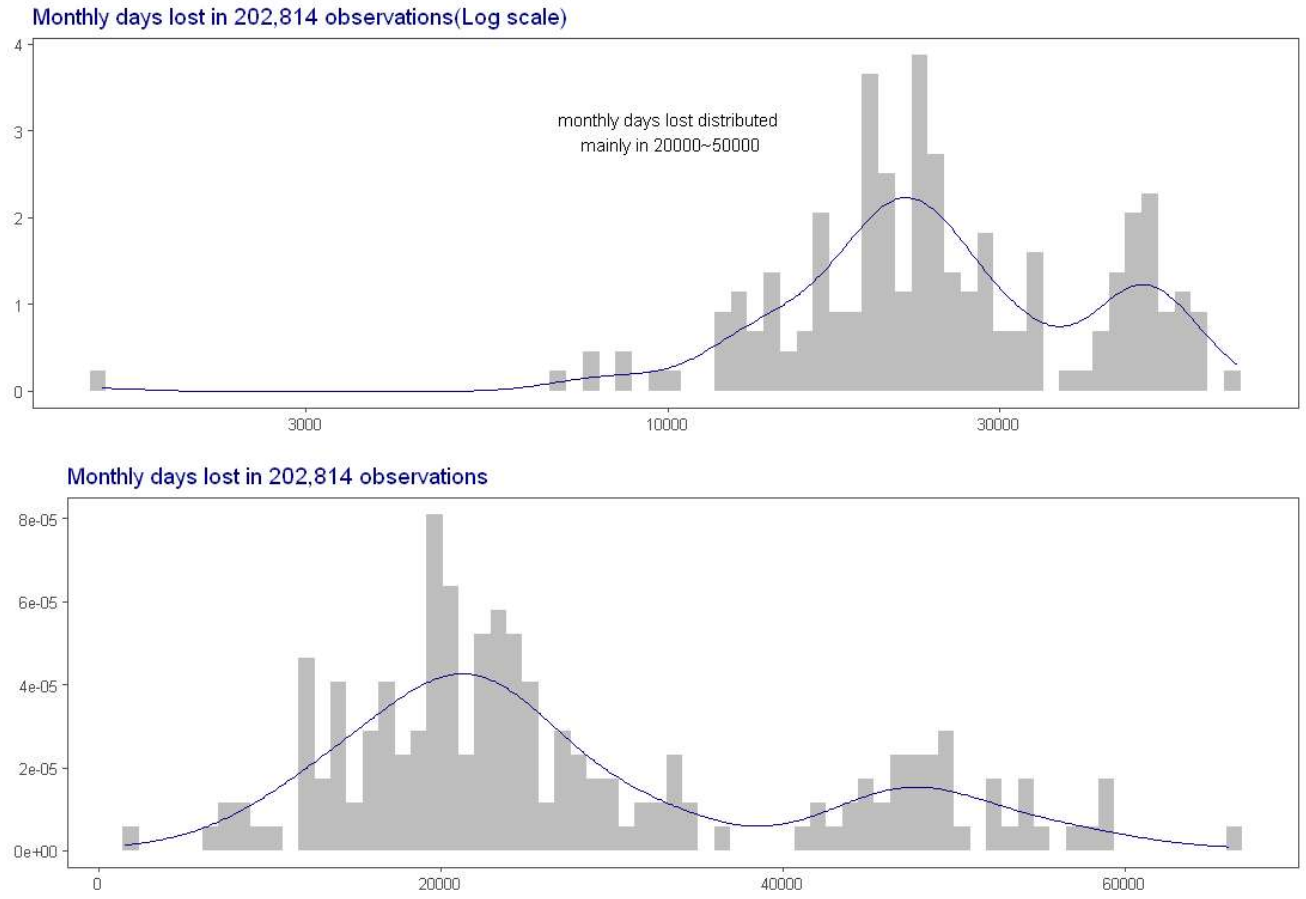


We can see the overall values of days lost varies more than that of days restrict.

The following histograms shows the monthly distribution of days lost sum by each month.

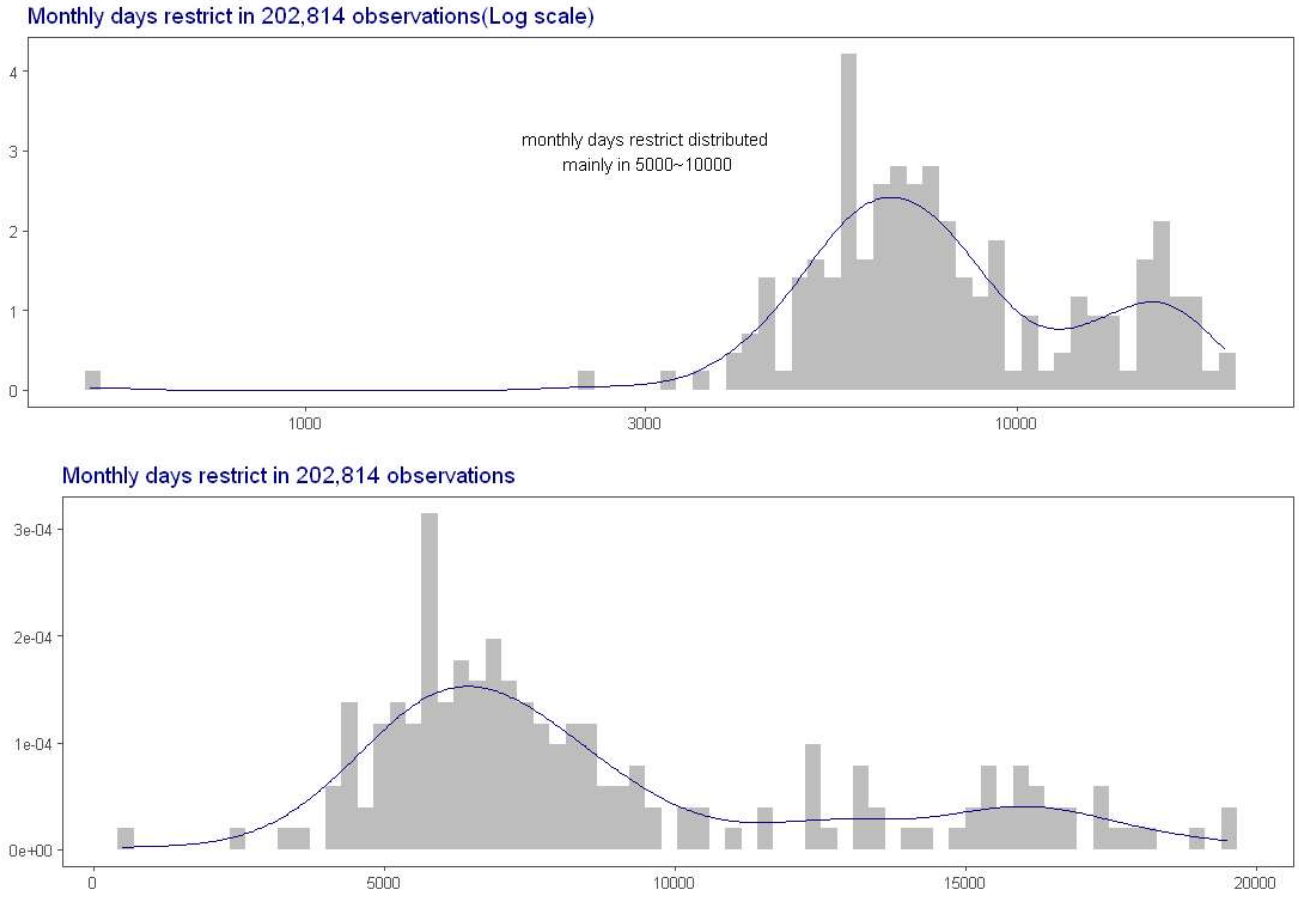
Hide

```
p1<-ggplot(us_data_by_days, aes(x=DAYS_LOST_SUM)) +
  geom_histogram(aes(y = ..density..),bins=70, fill = "grey")+
  geom_density(color="#000080")+
  scale_x_continuous(trans='log10')+
   labs(x = NULL, y = NULL) +
  annotate("text", x = 10000, y = 3, label = "monthly days lost distributed\n mainly in 20000
~50000") +
  ggtitle("Monthly days lost in 202,814 observations(Log scale)")+
  my_theme

p2<-ggplot(us_data_by_days, aes(x=DAYS_LOST_SUM)) +
  geom_histogram(aes(y = ..density..),bins=70, fill = "grey")+
  geom_density(color="#000080")+
   labs(x = NULL, y = NULL) +
  ggtitle("Monthly days lost in 202,814 observations")+
  my_theme
grid.arrange(p1,p2,ncol=1)
```



Hide

```
p1<-ggplot(us_data_by_days, aes(x=DAYS_RESTRICT_SUM)) +
  geom_histogram(aes(y = ..density..),bins=70, fill = "grey")+
  geom_density(color="#000080")+
  scale_x_continuous(trans='log10')+
   labs(x = NULL, y = NULL) +
  annotate("text", x = 3000, y = 3, label = "monthly days restrict distributed\n mainly in 50
00~10000") +
  ggtitle("Monthly days restrict in 202,814 observations(Log scale)")+
  my_theme

p2<-ggplot(us_data_by_days, aes(x=DAYS_RESTRICT_SUM)) +
  geom_histogram(aes(y = ..density..),bins=70, fill = "grey")+
  geom_density(color="#000080")+
   labs(x = NULL, y = NULL) +
  ggtitle("Monthly days restrict in 202,814 observations")+
  my_theme
grid.arrange(p1,p2,ncol=1)
```

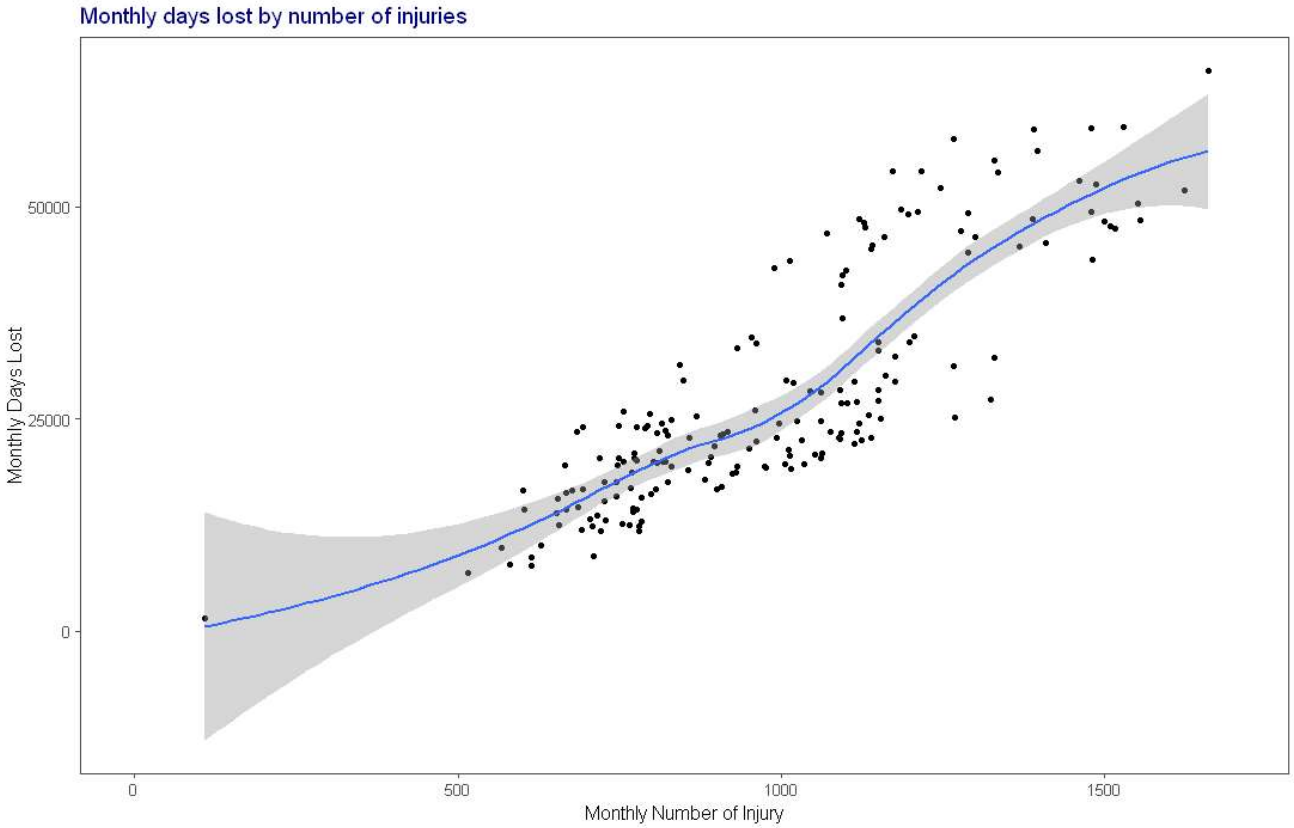


As expected, days lost and days restrict have similar distributions, and they are all highly right-skewed.

Analyze the monthly relationship between days lost and number of injuries

Hide

```
us_data_by_days_with_injury <- us_data[, c('ACCIDENT_DT','DAYS_LOST', 'NO_INJURIES')] %>%
  group_by(year(ACCIDENT_DT), month(ACCIDENT_DT)) %>%
  summarise(DAYS_LOST_SUM = sum(DAYS_LOST),NO_INJURIES_SUM=sum(NO_INJURIES))

us_data_by_days_with_injury <- set_col_name(us_data_by_days_with_injury,1)

ggplot(us_data_by_days_with_injury, aes(x=NO_INJURIES_SUM, y=DAYS_LOST_SUM))+
  geom_point()+
  geom_smooth()+
  labs(x = "Monthly Number of Injury", y = "Monthly Days Lost") +
  scale_x_continuous(limits = c(0,1700))+
  ggtitle("Monthly days lost by number of injuries")+
  my_theme
```

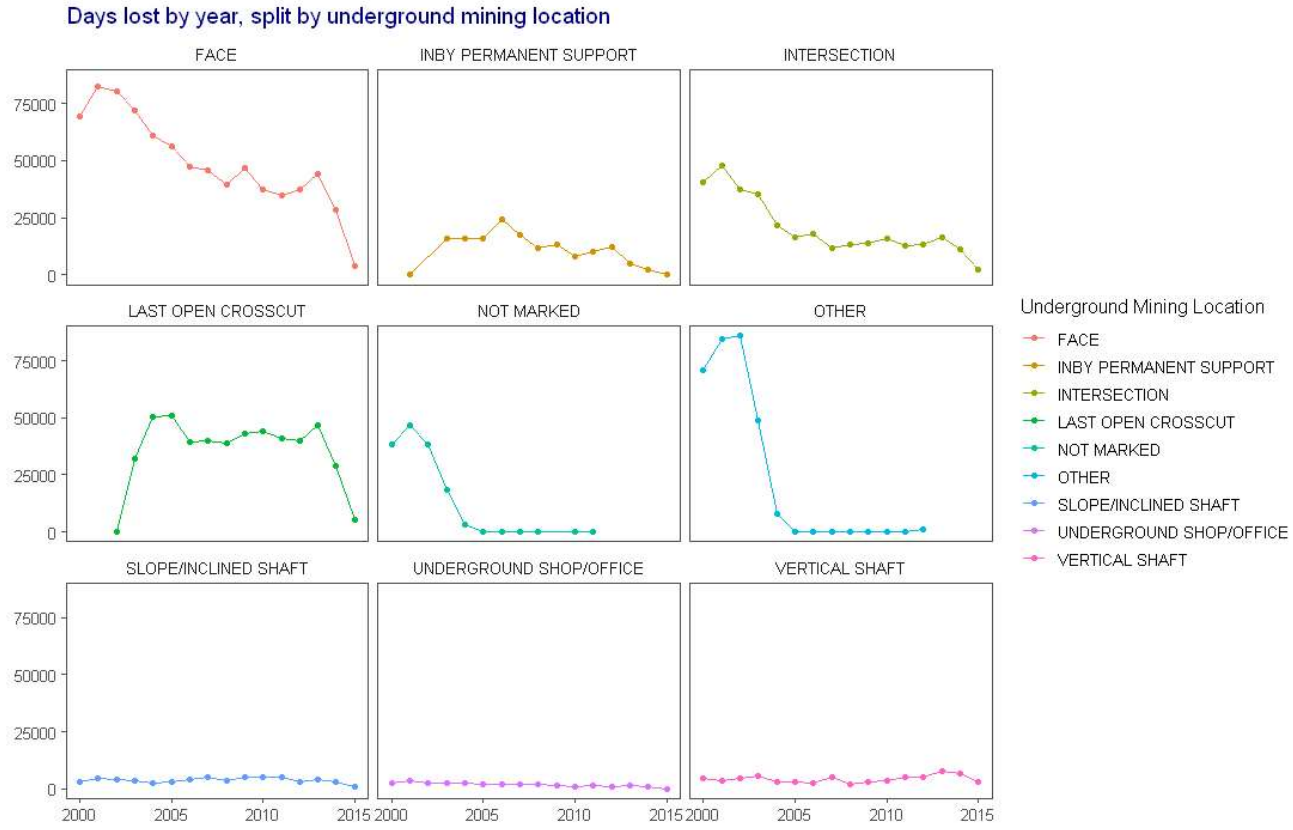


With the increasing of the number of injuries, days lost becomes larger.

Analyze the number of days lost by year for each underground location. We'll use facet can split each underground location up by drawing each of them in one plot separately

Hide

```
#initially transform data
us_data_by_days_with_loc <- us_data[, c('ACCIDENT_DT','DAYS_LOST', 'UG_LOCATION')] %>%
  group_by(year(ACCIDENT_DT),  UG_LOCATION) %>%
  summarise(DAYS_LOST_SUM = sum(DAYS_LOST))
  us_data_by_days_with_loc <- set_col_name(us_data_by_days_with_loc)



us_data_by_days_with_loc<-set_remove_NA(us_data_by_days_with_loc, "UG_LOCATION")


ggplot(us_data_by_days_with_loc, aes(x=ACCIDENT_YEAR, y=DAYS_LOST_SUM, color = UG_LOCATION))
 +
  facet_wrap(~UG_LOCATION,nrow = 4) +
  geom_point()+
  geom_line()+
  labs(x = NULL, y = NULL, color="Underground Mining Location") +
  ggtitle("Days lost by year, split by underground mining location")+
  my_theme
```

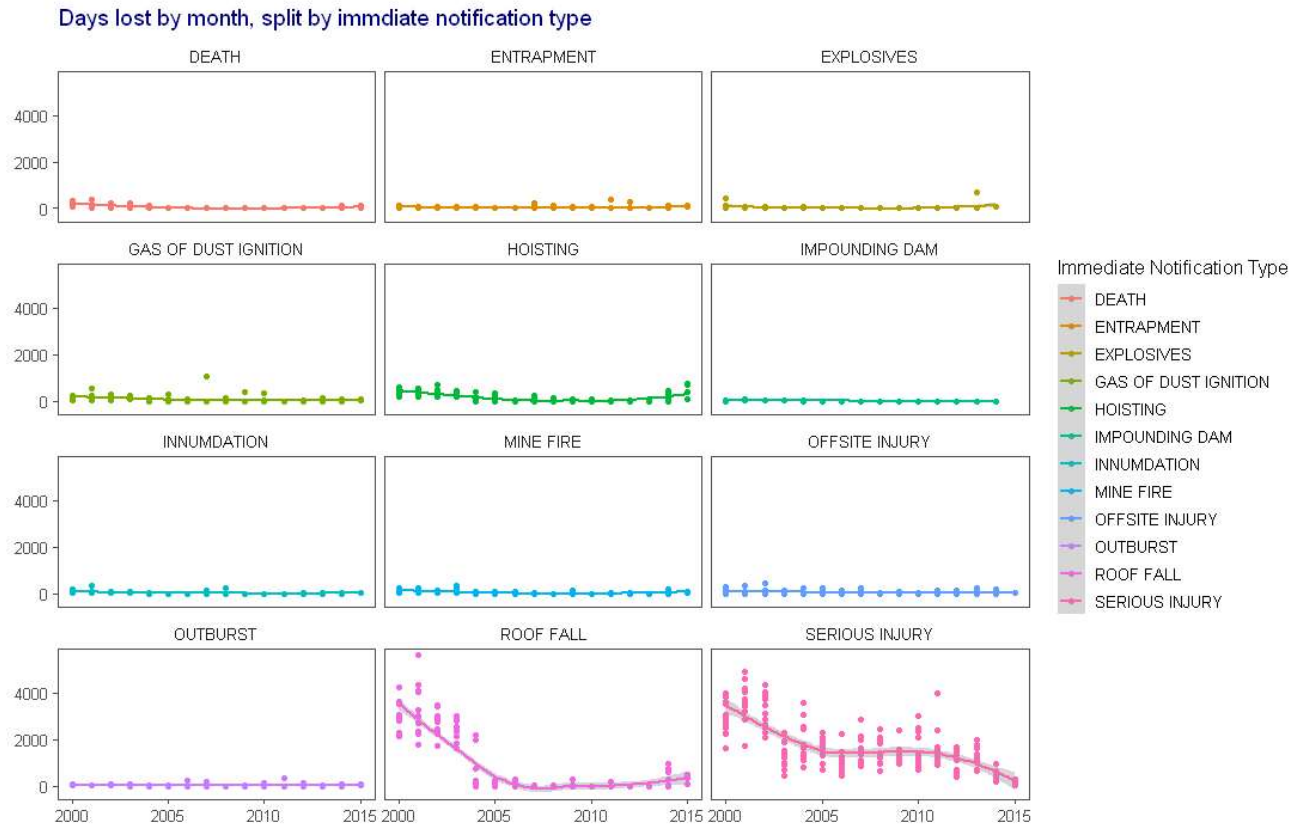


Different underground locations show different days lost trend by year.

Analyze the number of days lost by month for each type of immediate notification.

Hide

```
#initially transform data
us_data_by_days_with_not <- us_data[, c('ACCIDENT_DT','DAYS_LOST', 'IMMED_NOTIFY')] %>%
  group_by(year(ACCIDENT_DT), month(ACCIDENT_DT), IMMED_NOTIFY) %>%
  summarise(DAYS_LOST_SUM = sum(DAYS_LOST))
  us_data_by_days_with_not <- set_col_name(us_data_by_days_with_not,1)

us_data_by_days_with_not<-set_remove_NA(us_data_by_days_with_not,"IMMED_NOTIFY",1)

ggplot(us_data_by_days_with_not, aes(x=ACCIDENT_YEAR, y=DAYS_LOST_SUM, color = IMMED_NOTIFY))
+
  facet_wrap(~IMMED_NOTIFY,nrow = 4) +
  geom_point()+
  geom_smooth()+
  labs(x = NULL, y = NULL, color="Immediate Notification Type") +
  ggtitle("Days lost by month, split by immdiate notification type")+
  my_theme
```

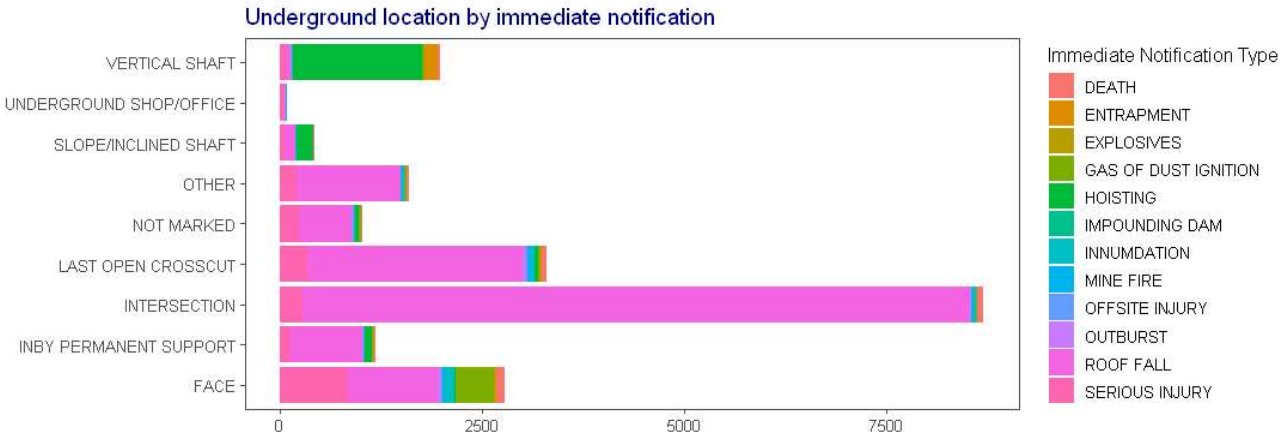


Only a few type (ROOF FALL, SERIOUS INJURY) have a relatively longer days lost, the overall trend is decreasing during the time.

use underground location and immediate notification to visualize the distribution of accident amount

Hide

```
location_by_method_count <- select(us_data, UG_LOCATION, IMMED_NOTIFY)

location_by_method_count$IMMED_NOTIFY<-ifelse(location_by_method_count$IMMED_NOTIFY == "NO VA
LUE FOUND"| location_by_method_count$IMMED_NOTIFY == "NOT MARKED" , NA, location_by_method_co
unt$IMMED_NOTIFY)
location_by_method_count$UG_LOCATION<-ifelse(location_by_method_count$UG_LOCATION == "NO VALU
E FOUND", NA, location_by_method_count$UG_LOCATION)
location_by_method_count<- na.omit(location_by_method_count)

ggplot(location_by_method_count, aes(y=UG_LOCATION))+
  geom_bar(aes(fill=IMMED_NOTIFY))+
  labs(x = NULL, y = NULL, fill="Immediate Notification Type") +
  ggtitle("Underground location by immediate notification")+
  my_theme
```



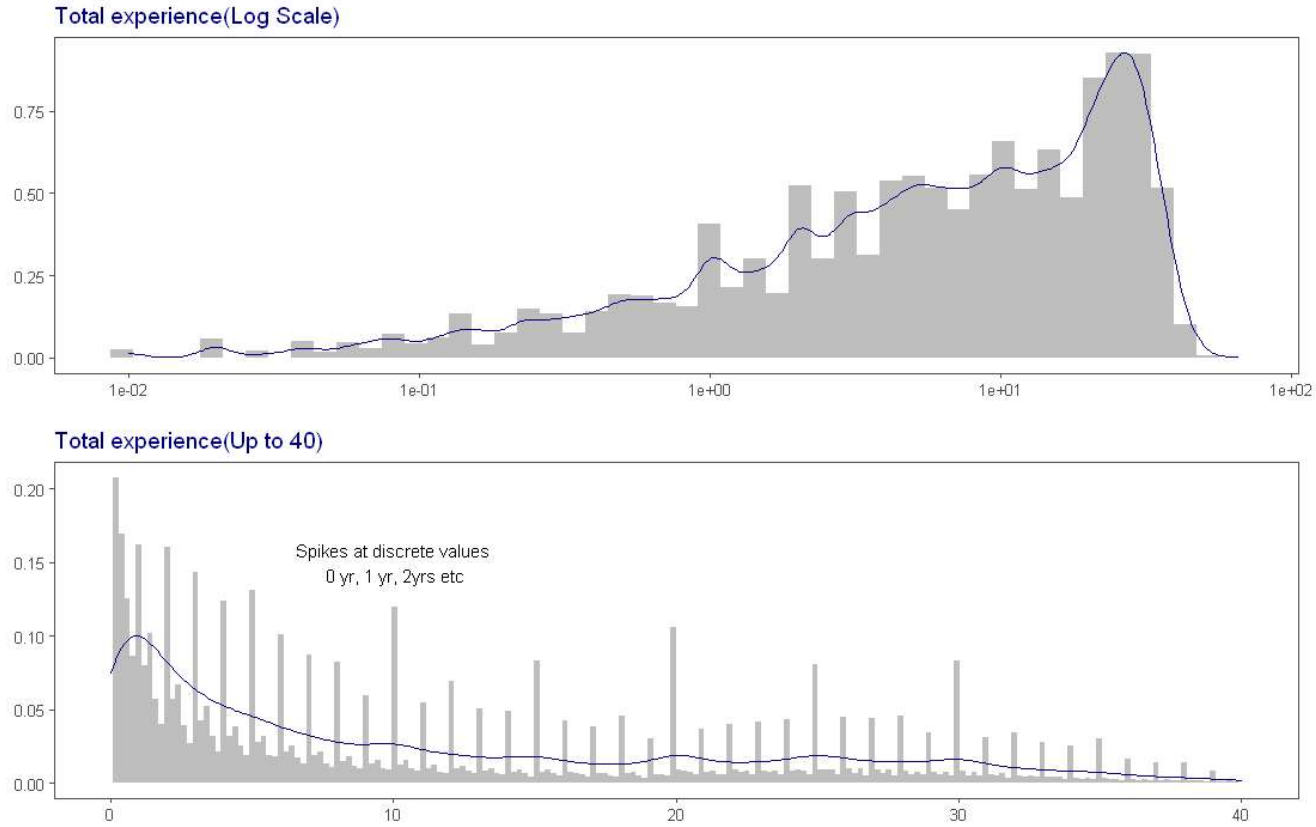
As expected, the immediate notification types which have more days lost in the previous graph have more account amount than others, and there is also a clear relationship between underground location and immediate notification.

Analyzing experience data

we'll use two histograms (log10 scale and truncated linear scale) to analyze the these variables separately, as they are right-skewed

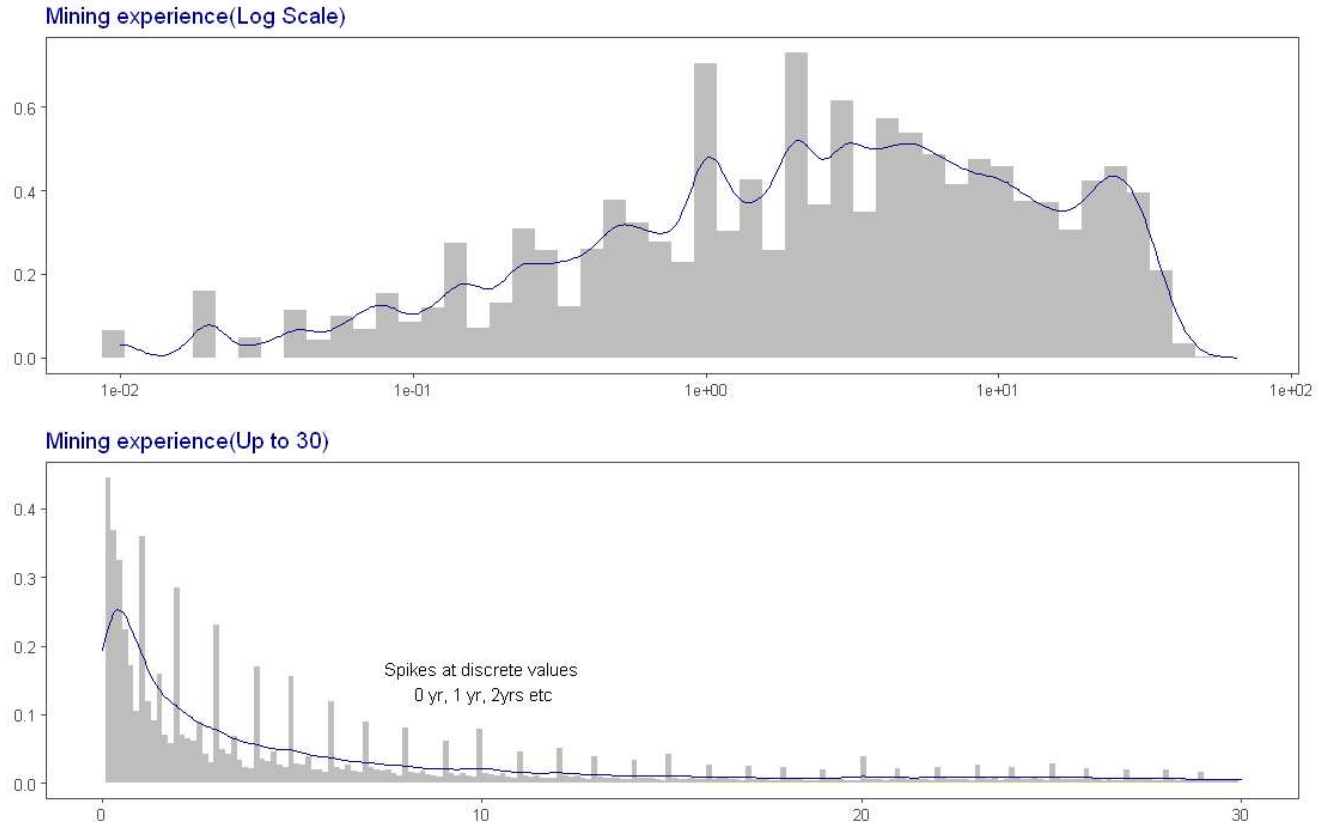
Hide

```
p1 <- ggplot(us_data, aes(x = TOT_EXPER)) +
  geom_histogram(aes(y = ..density..),bins=50, fill = "grey") +
  geom_density(colour = "#000080") +
  scale_x_continuous(trans='log10')+
    labs(x = NULL, y = NULL) +
  ggtitle("Total experience(Log Scale)") +
  my_theme

p2 <- ggplot(us_data, aes(x = TOT_EXPER)) +
  geom_histogram(aes(y = ..density..),bins=200, fill = "grey") +
  geom_density(colour = "#000080") +
  scale_x_continuous(limits=c(0, 40))+
  annotate("text", x = 10, y = 0.15, label = "Spikes at discrete values\n 0 yr, 1 yr, 2yrs et
c") +
    labs(x = NULL, y = NULL) +
  ggtitle("Total experience(Up to 40)") +
  my_theme

grid.arrange(p1, p2, ncol=1)
```



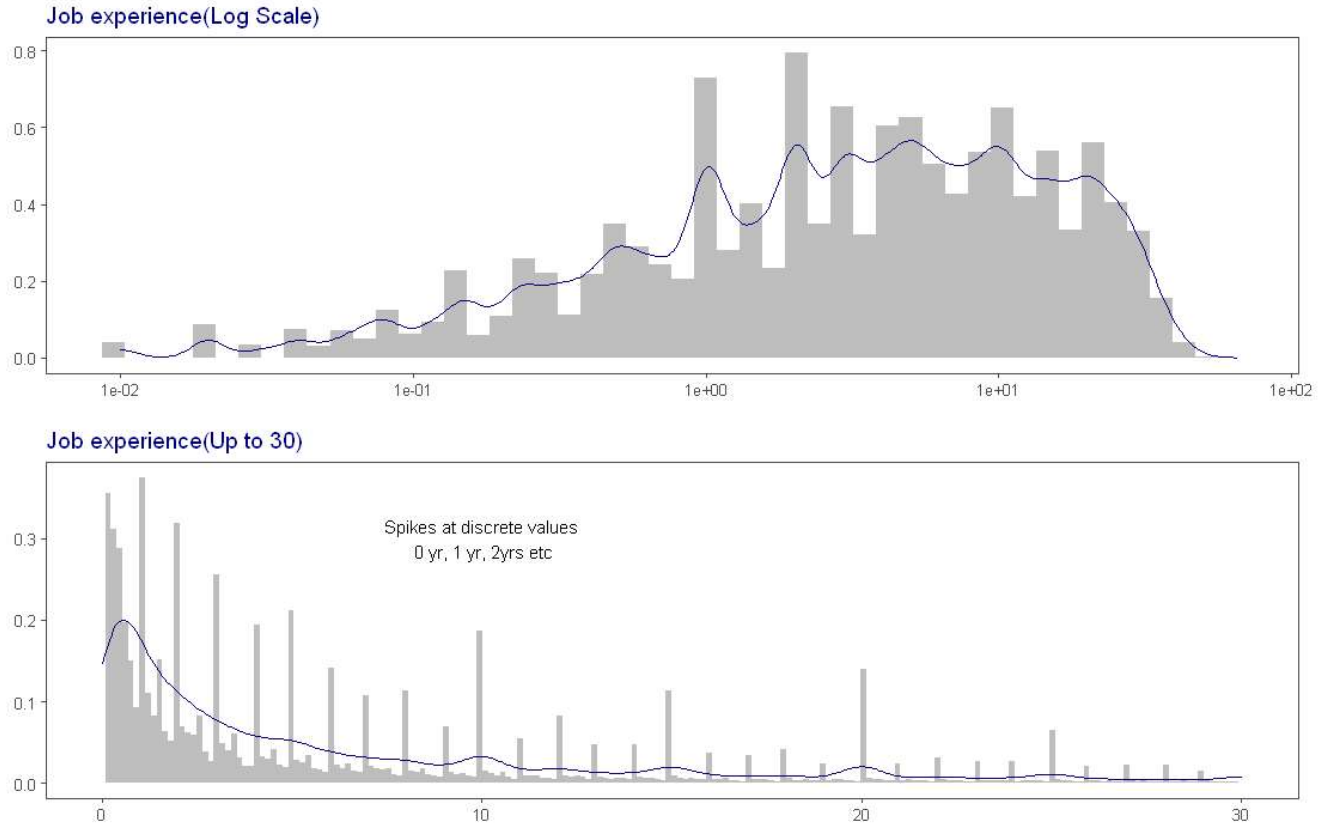
Hide

```
p1 <- ggplot(us_data, aes(x = MINE_EXPER)) +
  geom_histogram(aes(y = ..density..),bins=50, fill = "grey") +
  geom_density(colour = "#000080") +
  scale_x_continuous(trans='log10')+
  labs(x = NULL, y = NULL) +
  ggtitle("Mining experience(Log Scale)") +
  my_theme

p2 <- ggplot(us_data, aes(x = MINE_EXPER)) +
  geom_histogram(aes(y = ..density..),bins=200, fill = "grey") +
  geom_density(colour = "#000080") +
  scale_x_continuous(limits=c(0, 30))+
  annotate("text", x = 10, y = 0.15, label = "Spikes at discrete values\n 0 yr, 1 yr, 2yrs et
c") +
    labs(x = NULL, y = NULL) +
  ggtitle("Mining experience(Up to 30)") +
  my_theme

grid.arrange(p1, p2, ncol=1)
```



Hide

```
p1 <- ggplot(us_data, aes(x = JOB_EXPER)) +
  geom_histogram(aes(y = ..density..),bins=50, fill = "grey") +
  geom_density(colour = "#000080") +
  scale_x_continuous(trans='log10')+
  labs(x = NULL, y = NULL) +
  ggtitle("Job experience(Log Scale)") +
  my_theme

p2 <- ggplot(us_data,aes(x = JOB_EXPER)) +
  geom_histogram(aes(y = ..density..),bins=200, fill = "grey") +
  geom_density(colour = "#000080") +
  scale_x_continuous(limits=c(0, 30))+
  annotate("text", x = 10, y = 0.3, label = "Spikes at discrete values\n 0 yr, 1 yr, 2yrs et
c") +
  labs(x = NULL, y = NULL) +
  ggtitle("Job experience(Up to 30)") +
  my_theme

grid.arrange(p1, p2, ncol=1)
```



total experience, mine experience and job experience were all having similar distributions with each other. Values with low experience took a large proportion in their distributions.

To perform the following analyses, we need to clean and transform the experience data first

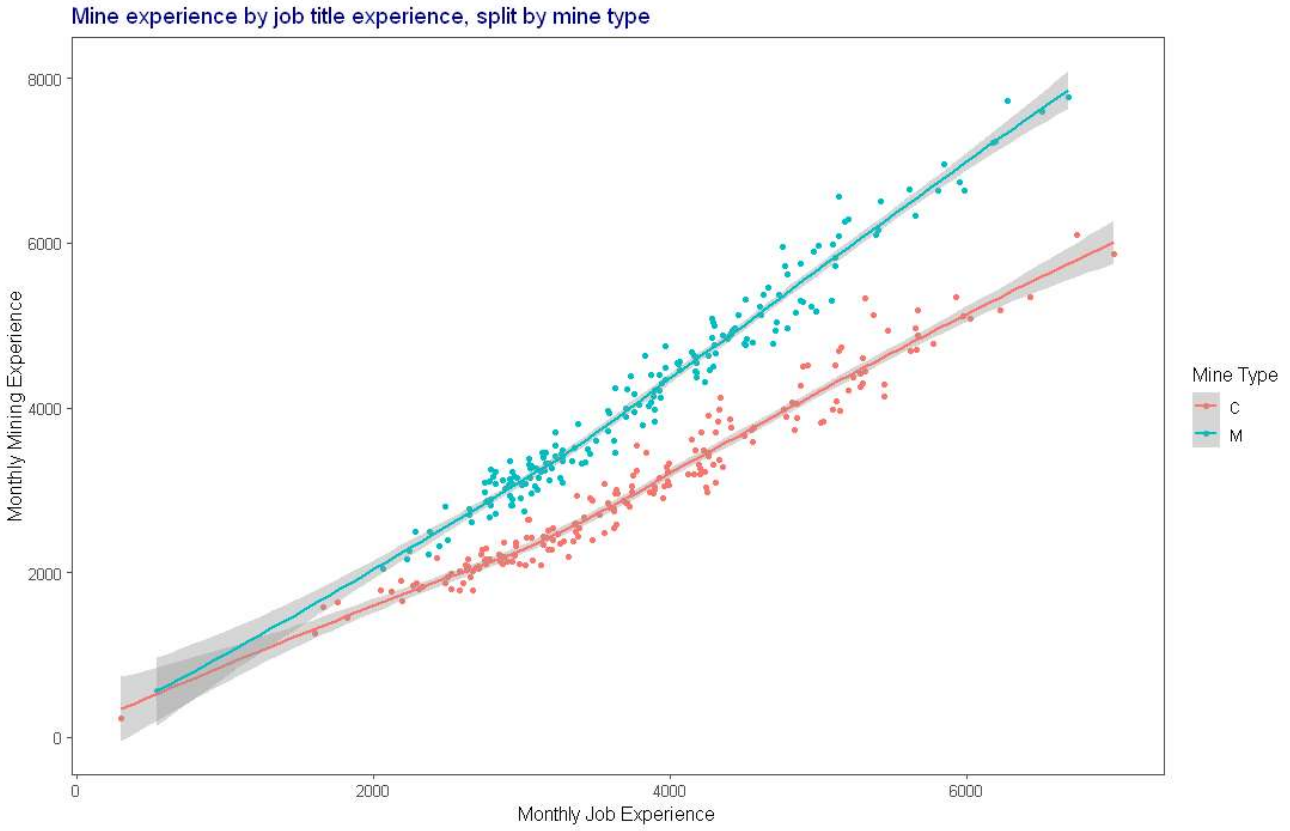
Hide

```
#transform data
us_data_with_exp <- select(us_data, ACCIDENT_DT, TOT_EXPER, MINE_EXPER, JOB_EXPER,NO_INJURIE
S, COAL_METAL_IND, SUBUNIT)
varlist<- setdiff(colnames(us_data_with_exp),c("ACCIDENT_DT","COAL_METAL_IND","SUBUNIT"))
treatment_plan <- design_missingness_treatment(us_data_with_exp, varlist=varlist)
us_data_with_exp <- prepare(treatment_plan, us_data_with_exp)
```

The following analyses the monthly relationship between job experience and mine experience split by different types of mine.

Hide

```
us_data_with_exp_type <- us_data_with_exp[, c('ACCIDENT_DT', 'MINE_EXPER','JOB_EXPER','COAL_M
ETAL_IND')] %>%
  group_by(year(ACCIDENT_DT), month(ACCIDENT_DT), COAL_METAL_IND) %>%
  summarise(MINE_EXPER_SUM=sum(MINE_EXPER), JOB_EXPER_SUM=sum(JOB_EXPER))
  us_data_with_exp_type <-set_col_name(us_data_with_exp_type,1)


ggplot(us_data_with_exp_type, aes(x=JOB_EXPER_SUM, y=MINE_EXPER_SUM, color=COAL_METAL_IND))+
  geom_point()+
  geom_smooth()+
  labs(x="Monthly Job Experience", y="Monthly Mining Experience", color="Mine Type")+
  ggtitle("Mine experience by job title experience, split by mine type")+
  my_theme
```

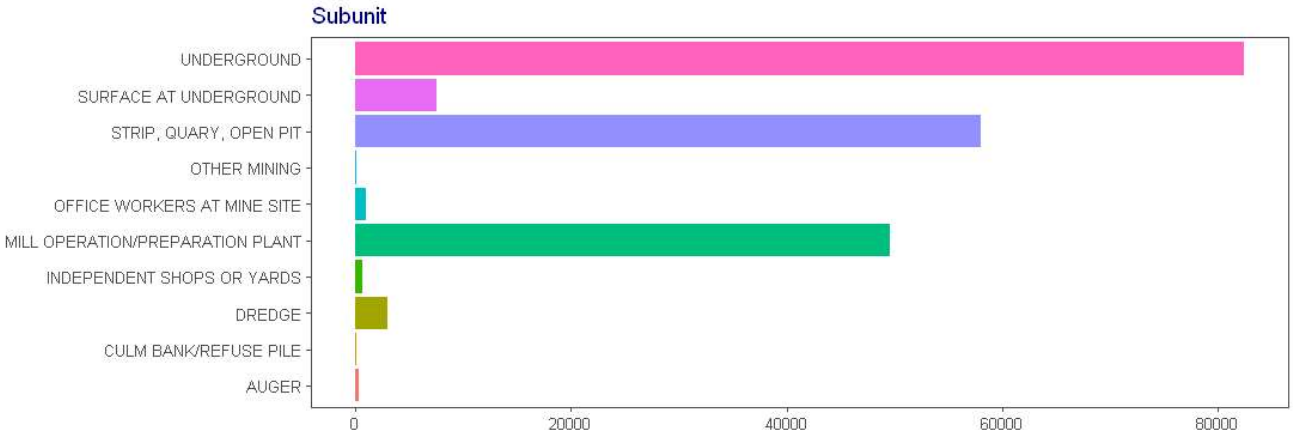


mine experience has a nearly linear relationship with job title experience overall, and people affected by accidents in coal mine tend to have less mine experience than people affected by accidents in metal/non-metal mine when they have similar job title experience.

The following graph shows the accident amount in each subunit

Hide

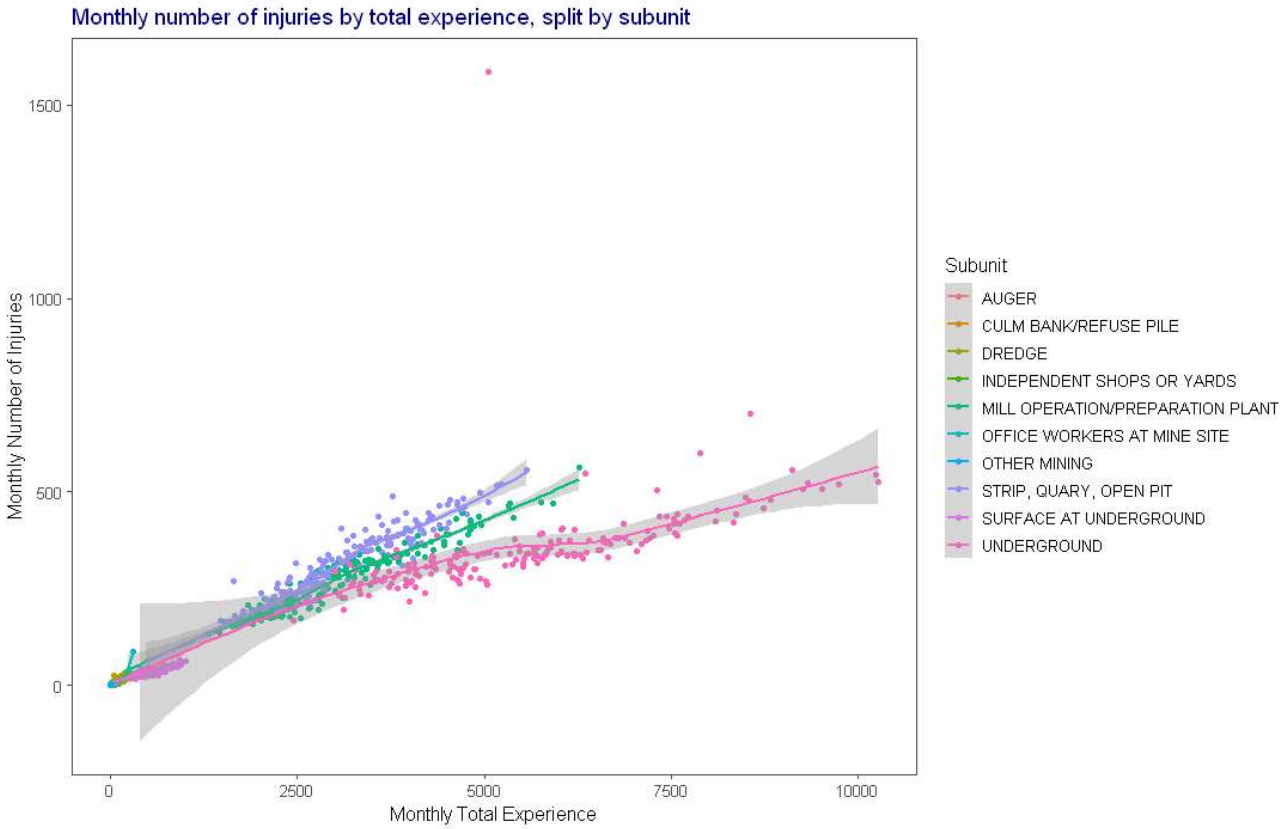
```
ggplot(us_data, aes(x = SUBUNIT, fill = SUBUNIT)) +
  geom_bar(show.legend = FALSE) +
  labs(x = NULL, y = NULL) +
  coord_flip() +
  ggtitle("Subunit")+
  my_theme
```



Then we'll analyze the monthly relationship between total experience and number of injuries split by subunit

Hide

```
us_data_with_exp_injury <- us_data_with_exp[, c('ACCIDENT_DT', 'TOT_EXPER','NO_INJURIES', 'SU
BUNIT')] %>%
  group_by(year(ACCIDENT_DT), month(ACCIDENT_DT), SUBUNIT) %>%
  summarise(TOT_EXPER_SUM=sum(TOT_EXPER), NO_INJURIES_SUM=sum(NO_INJURIES))
  us_data_with_exp_injury <-set_col_name(us_data_with_exp_injury,1)


ggplot(us_data_with_exp_injury, aes(x=TOT_EXPER_SUM, y=NO_INJURIES_SUM, color=SUBUNIT))+
  geom_point()+
  geom_smooth()+
  labs(x="Monthly Total Experience",y="Monthly Number of Injuries",color="Subunit")+
  ggtitle("Monthly number of injuries by total experience, split by subunit")+
  my_theme
```

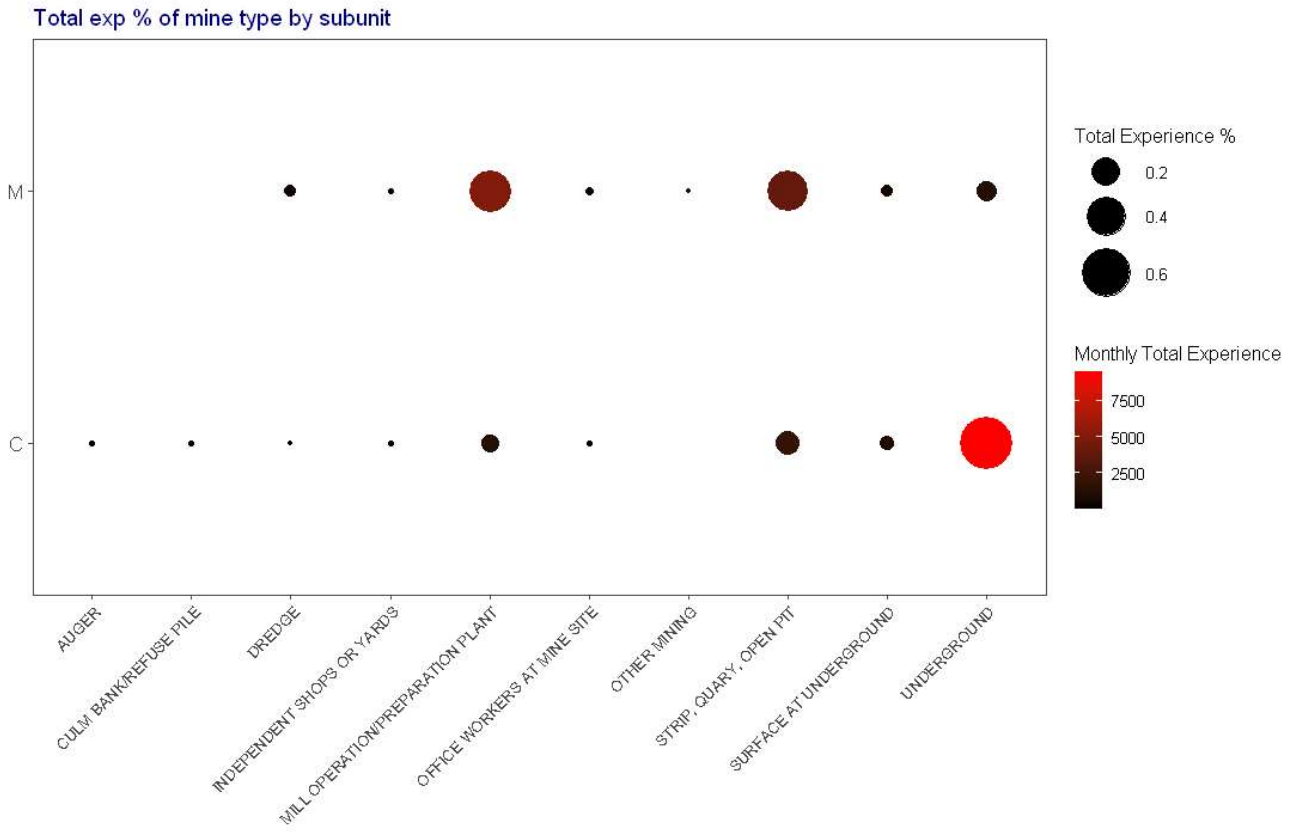


we can see that there are three main subunits here have more monthly injuries than others, and the relationships between monthly number of injuries and monthly total experience are similar among these subunits.

Use mine type and subunit to visualize the distribution of experience percent and amount to see if the subunit and mine type are related. We can use geom_count as it can show the relationship between two categorical variables and can also add a continuous variable (TOT_EXPER in this case) to present its size and amount in each point.

Hide

```
us_data_with_exp_subunit_mine <- us_data_with_exp[, c('ACCIDENT_DT', 'TOT_EXPER','SUBUNIT',
'COAL_METAL_IND')] %>%
  group_by(year(ACCIDENT_DT), month(ACCIDENT_DT),SUBUNIT,COAL_METAL_IND) %>%
  summarise(TOT_EXPER_SUM=sum(TOT_EXPER),TOT_EXPER_MEAN=mean(TOT_EXPER),TOT_EXPER_COUNT=sum(T
OT_EXPER*0 + 1))

us_data_with_exp_subunit_mine <-set_col_name(us_data_with_exp_subunit_mine,1)


us_data_with_exp_mine <- us_data_with_exp_subunit_mine %>% group_by(COAL_METAL_IND) %>% summa
rise(EXPER_COUNT = sum(TOT_EXPER_COUNT))

us_data_with_exp_subunit_mine<-merge(us_data_with_exp_subunit_mine, us_data_with_exp_mine, by
='COAL_METAL_IND')

us_data_with_exp_subunit_mine$TOT_EXPER_PERCENT <- us_data_with_exp_subunit_mine$TOT_EXPER_CO
UNT/us_data_with_exp_subunit_mine$EXPER_COUNT * 100

ggplot(us_data_with_exp_subunit_mine, aes(x = SUBUNIT, y = COAL_METAL_IND)) +
  geom_count(aes(size=TOT_EXPER_PERCENT, color=TOT_EXPER_SUM)) +
  ggtitle("Total exp % of mine type by subunit") +
  scale_color_continuous(low = "black", high = "red") +

  scale_size_continuous(range = c(0.1, 15)) +
  labs(size="Total Experience %", color="Monthly Total Experience")+
  my_theme+
    theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 10),
        axis.text.y = element_text(angle =  0, size = 12),
        axis.title.x = element_blank(),
        axis.title.y = element_blank())
```



We can see that people affected in coal mine were experienced mainly in ‘UNDERGROUND’ subunit, and people affected in metal/non-metal mine were experienced mainly in ‘MILL OPERATION/PERPARATION PLANT’ and ‘STRIP, QUARY, OPEN PIT’