Project1

Daniel Lassiter, Tom Muhlbaur, and Rachael Stryker

9/28/2020

Generating Hypotheses:

The variables that we decided to hone in on for our exploratory data analysis are:

- Quantitative: CARS (# cars w/hazmat), CARSDMG (# hazmat cars that were damaged or derailed), CARSHZD (# of cars that released hazmat), MONTH, DAY, TIMEHR, TIMEMIN, AMPM, TRNSPD, HIGHSPD
- Qualitative: RR2 (second railroad involved), TYPE (accident type), TYPEQ (car type), Cause (manually assigned from CAUSE), STATION, WEATHER, VISIBLTY

Loading libraries and data, settign directories, and processing data

```
#Import libraries
library(ggplot2)
library(lattice)

#Set directories:
setwd('..')
wd <- getwd()

sourcedir <- paste0(wd, "/Source/")
traindir <-  paste0(wd, "/Data/TrainData")


# Source AccidentInput
setwd(sourcedir)
source("AccidentInput.R")
source("SPM_Panel.R")
source("PCAplots.R")

# Create a list of data frames for each year of accident data

acts <- file.inputl(traindir)

# Create a data frame with all accidents from all years from 2001 - 2019
# with columns that are consistent for all of these years

# Get a common set the variables
```

```
comvar <- intersect(colnames(acts[[1]]), colnames(acts[[8]]))

# the combined data frame

totacts <- combine.data(acts)

# Update the TYPE variable to contain more legible values

totacts$TYPE <- factor(totacts$TYPE, labels = c("Derailment", "HeadOn", "Rearend", "Side", "Raking", "Bi

# Update WEATHER variable:

totacts$WEATHER <- factor(totacts$WEATHER, labels = c("clear", "cloudy", "rain", "fog", "sleet", "snow"]

# Update the visibility variable

totacts$VISIBLTY <- factor(totacts$VISIBLTY, labels = c("dawn", "day", "dusk", "dark"))

# Update the cause variable to have more legible values

# Accident cause
totacts$Cause <- rep(NA, nrow(totacts))

totacts$Cause[which(substr(totacts$CAUSE, 1, 1) == "M")] <- "(M) Miscellaneous Causes Not Otherwise Lis
totacts$Cause[which(substr(totacts$CAUSE, 1, 1) == "T")] <- "(T) Rack, Roadbed and Structures"
totacts$Cause[which(substr(totacts$CAUSE, 1, 1) == "S")] <- "(S) Signal and Communication"
totacts$Cause[which(substr(totacts$CAUSE, 1, 1) == "H")] <- "(H) Train operation - Human Factors"
totacts$Cause[which(substr(totacts$CAUSE, 1, 1) == "E")] <- "(E) Mechanical and Electrical Failures"

# This new variable, Cause, has to be a factor
totacts$Cause <- factor(totacts$Cause)
```

Looking at casualties

Create a data frame containing only accidents with one or more casualties. Use the variables "INCDTNO", "YEAR", "MONTH", "DAY", "TIMEHR", "TIMEMIN" to determine if there are duplicates in the accident reports with one or more casualties. Report number of duplicates.Show a box plot of casualities per accident per year.

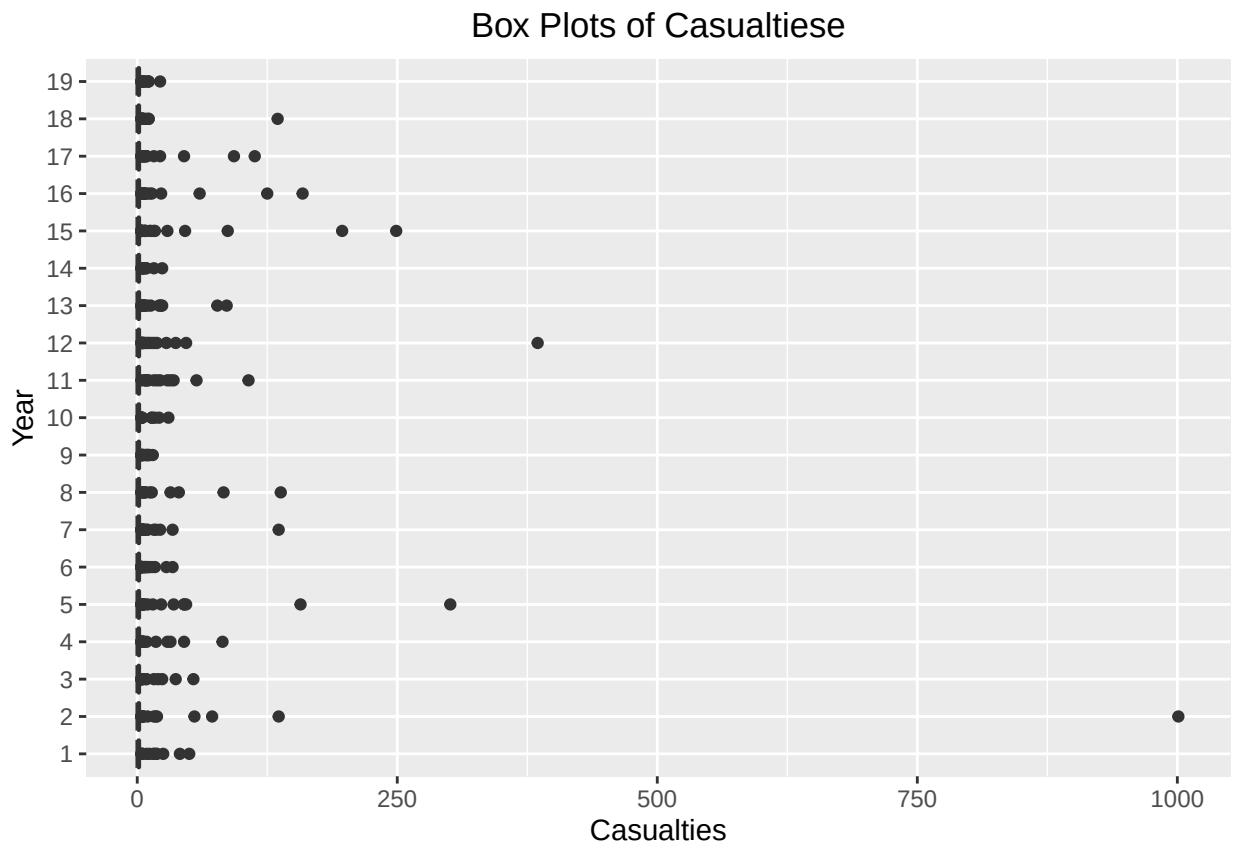
```
totacts$Casualty <- totacts$TOTINJ + totacts$TOTKLD
totacts_wCasualties <- subset.data.frame(totacts, totacts$Casualty>0)
totacts_wCasualties_DR <- totacts_wCasualties[!(duplicated(totacts_wCasualties[, c("INCDTNO", "YEAR", "

#Reset rownames (observation #s) for sequential numbering- otherwise they will remain the #s from totac
rownames(totacts_wCasualties_DR) <- NULL

plt <- ggplot(data = totacts_wCasualties_DR, aes(x = as.factor(YEAR), y = Casualty)) +
  geom_boxplot() +
  coord_flip() +
  scale_fill_grey(start = 0.5, end = 0.8) +
  theme(plot.title = element_text(hjust = 0.5)) +
```

```r
  ggtitle("Box Plots of Casualtiese") +
  labs(x = "Year", y = "Casualties")


ggplot_build(plt)$data
```

```
## [[1]]
##    ymin lower middle upper ymax
## 1     1     1      1     2    3
## 2     1     1      1     2    3
## 3     1     1      1     2    3
## 4     1     1      1     2    3
## 5     1     1      1     2    3
## 6     1     1      1     2    3
## 7     1     1      1     2    3
## 8     1     1      1     2    3
## 9     1     1      1     2    3
## 10    1     1      1     2    3
## 11    1     1      1     2    3
## 12    1     1      1     2    3
## 13    1     1      1     2    3
## 14    1     1      1     2    3
## 15    1     1      1     2    3
## 16    1     1      1     2    3
## 17    1     1      1     2    3
## 18    1     1      1     2    3
## 19    1     1      1     2    3
##                                                                                 outliers
## 1                                      50, 41, 9, 16, 25, 18, 4, 4, 16, 19, 12, 5, 5
## 2                        10, 1001, 4, 5, 4, 72, 4, 18, 5, 55, 6, 4, 19, 16, 136, 6, 4
## 3                                        5, 37, 8, 4, 4, 9, 16, 24, 4, 5, 54, 20, 4
## 4                                  4, 9, 18, 45, 29, 5, 6, 4, 82, 6, 4, 5, 4, 32, 6, 4
## 5                  4, 5, 45, 4, 10, 23, 5, 35, 5, 4, 6, 15, 157, 4, 301, 45, 4, 47, 7
## 6  11, 4, 4, 8, 9, 4, 11, 7, 4, 4, 28, 17, 4, 7, 4, 4, 4, 5, 6, 8, 4, 34, 5, 4, 14
## 7                   4, 5, 9, 4, 6, 6, 22, 5, 4, 5, 6, 18, 136, 4, 6, 34, 10, 4, 4, 16
## 8                                     12, 14, 32, 40, 6, 4, 83, 7, 138, 8, 7, 4, 4
## 9                        4, 11, 9, 4, 15, 10, 9, 5, 5, 7, 11, 12, 4, 6, 4, 6, 4, 9, 4
## 10                                          14, 4, 4, 4, 17, 30, 5, 21, 4, 14, 4
## 11                                 16, 9, 35, 22, 57, 7, 11, 9, 29, 4, 19, 32, 107, 8
## 12             5, 17, 4, 10, 385, 19, 6, 15, 37, 5, 5, 28, 4, 11, 5, 8, 4, 47, 13, 7
## 13                       4, 24, 77, 4, 10, 5, 7, 22, 4, 13, 5, 4, 23, 86, 7, 7, 21, 7
## 14                                             24, 7, 8, 6, 9, 16, 6, 5, 4, 4, 5
## 15     4, 4, 13, 4, 4, 7, 249, 46, 8, 4, 16, 4, 29, 197, 17, 8, 87, 12, 6, 4, 4, 7
## 16                       12, 6, 4, 60, 4, 14, 4, 125, 7, 8, 7, 23, 4, 5, 5, 9, 159, 4
## 17                            93, 16, 22, 6, 8, 4, 7, 9, 4, 9, 7, 7, 5, 113, 4, 45, 4
## 18                                      11, 11, 5, 7, 6, 135, 5, 4, 4, 9, 5, 4, 4, 5
## 19                                         4, 22, 10, 4, 5, 4, 5, 6, 4, 11, 6, 11, 8
##    notchupper notchlower  x flipped_aes PANEL group ymin_final ymax_final
## 1    1.111723  0.8882771  1       FALSE     1     1          1         50
## 2    1.110352  0.8896480  2       FALSE     1     2          1       1001
## 3    1.112570  0.8874297  3       FALSE     1     3          1         54
## 4    1.104182  0.8958179  4       FALSE     1     4          1         82
## 5    1.105568  0.8944318  5       FALSE     1     5          1        301
```

```
## 6    1.106767  0.8932335  6       FALSE     1     6          1         34
## 7    1.105805  0.8941954  7       FALSE     1     7          1        136
## 8    1.123755  0.8762448  8       FALSE     1     8          1        138
## 9    1.128579  0.8714214  9       FALSE     1     9          1         15
## 10   1.123755  0.8762448 10       FALSE     1    10          1         30
## 11   1.123003  0.8769972 11       FALSE     1    11          1        107
## 12   1.129439  0.8705614 12       FALSE     1    12          1        385
## 13   1.116479  0.8835209 13       FALSE     1    13          1         86
## 14   1.134499  0.8655015 14       FALSE     1    14          1         24
## 15   1.126098  0.8739023 15       FALSE     1    15          1        249
## 16   1.131212  0.8687881 16       FALSE     1    16          1        159
## 17   1.122632  0.8773682 17       FALSE     1    17          1        113
## 18   1.124137  0.8758635 18       FALSE     1    18          1        135
## 19   1.119097  0.8809030 19       FALSE     1    19          1         22
##      xmin   xmax xid newx new_width weight colour  fill size alpha shape
## 1   0.625  1.375   1    1      0.75      1 grey20 white  0.5    NA    19
## 2   1.625  2.375   2    2      0.75      1 grey20 white  0.5    NA    19
## 3   2.625  3.375   3    3      0.75      1 grey20 white  0.5    NA    19
## 4   3.625  4.375   4    4      0.75      1 grey20 white  0.5    NA    19
## 5   4.625  5.375   5    5      0.75      1 grey20 white  0.5    NA    19
## 6   5.625  6.375   6    6      0.75      1 grey20 white  0.5    NA    19
## 7   6.625  7.375   7    7      0.75      1 grey20 white  0.5    NA    19
## 8   7.625  8.375   8    8      0.75      1 grey20 white  0.5    NA    19
## 9   8.625  9.375   9    9      0.75      1 grey20 white  0.5    NA    19
## 10  9.625 10.375  10   10      0.75      1 grey20 white  0.5    NA    19
## 11 10.625 11.375  11   11      0.75      1 grey20 white  0.5    NA    19
## 12 11.625 12.375  12   12      0.75      1 grey20 white  0.5    NA    19
## 13 12.625 13.375  13   13      0.75      1 grey20 white  0.5    NA    19
## 14 13.625 14.375  14   14      0.75      1 grey20 white  0.5    NA    19
## 15 14.625 15.375  15   15      0.75      1 grey20 white  0.5    NA    19
## 16 15.625 16.375  16   16      0.75      1 grey20 white  0.5    NA    19
## 17 16.625 17.375  17   17      0.75      1 grey20 white  0.5    NA    19
## 18 17.625 18.375  18   18      0.75      1 grey20 white  0.5    NA    19
## 19 18.625 19.375  19   19      0.75      1 grey20 white  0.5    NA    19
##    linetype
## 1     solid
## 2     solid
## 3     solid
## 4     solid
## 5     solid
## 6     solid
## 7     solid
## 8     solid
## 9     solid
## 10    solid
## 11    solid
## 12    solid
## 13    solid
## 14    solid
## 15    solid
## 16    solid
## 17    solid
## 18    solid
## 19    solid
```

```
plt
```



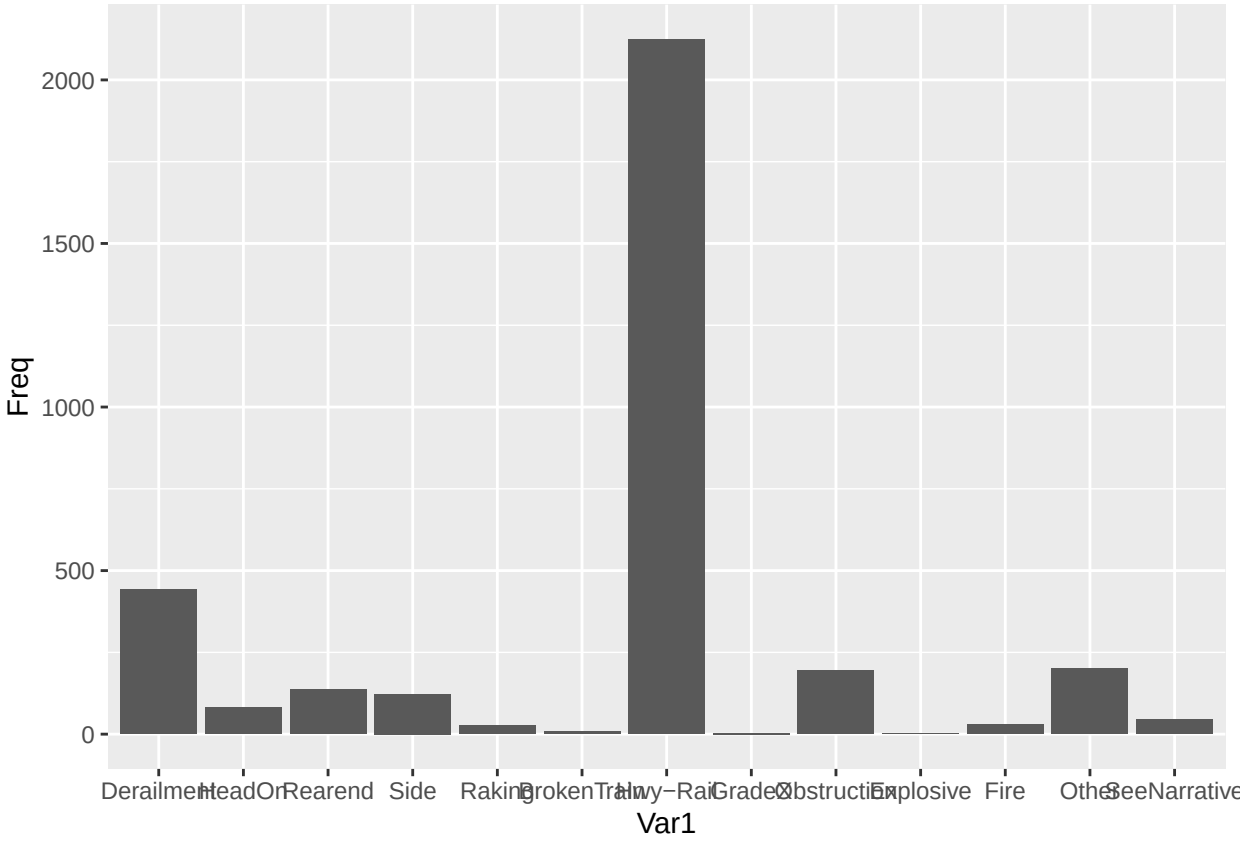

```
# ggpairs(totacts_wCasualties_DR[,c("cars", "EQPDMG", "ACCDMG", "TOTINJ", "TOTKLD")])
```

Make a barplot of the type of accidents for all accidents with one or more casualties.

```
library(ggplot2)

# Use table() to see the frequencies

table(totacts_wCasualties_DR$TYPE)
```

```
##
##   Derailment       HeadOn      Rearend         Side       Raking  BrokenTrain
##          442           82          136          123           26            8
##     Hwy-Rail       GradeX  Obstruction    Explosive         Fire        Other
##         2124            4          194            2           30          201
## SeeNarrative
##           45
```

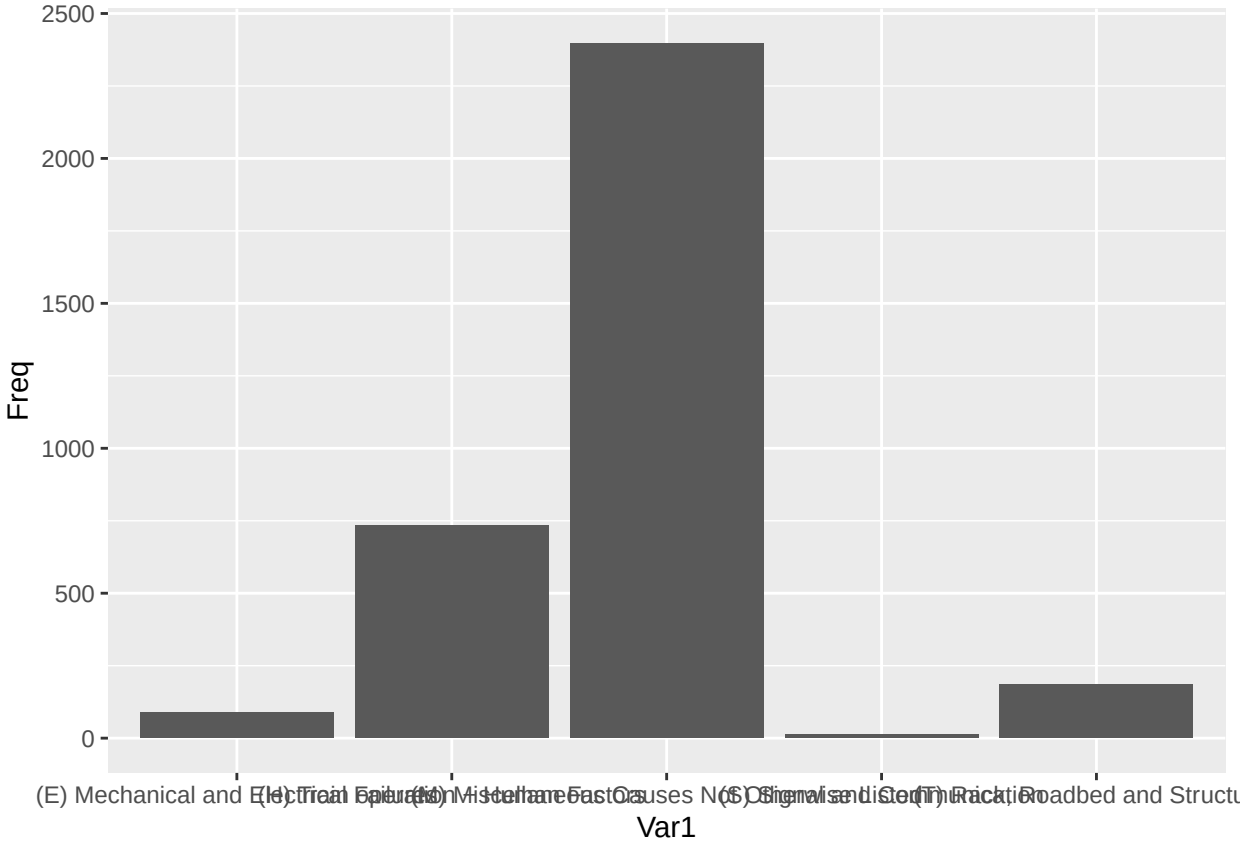
```
# Use barplot() to graph this

ggplot(as.data.frame(table(totacts_wCasualties_DR$TYPE)), aes(x = Var1, y= Freq)) + geom_bar(stat="iden
```



Highway-rail and derailments are the two accident types that most frequently yield one or more casualties.

Make a barplot of the cause of accidents for all accidents with one or more casualties.

```
# Use table() to see the frequencies

tbl <- as.data.frame(table(totacts_wCasualties_DR$Cause))

# Use barplot() to graph this

ggplot(tbl, aes(x = Var1, y= Freq)) + geom_bar(stat="identity")
```



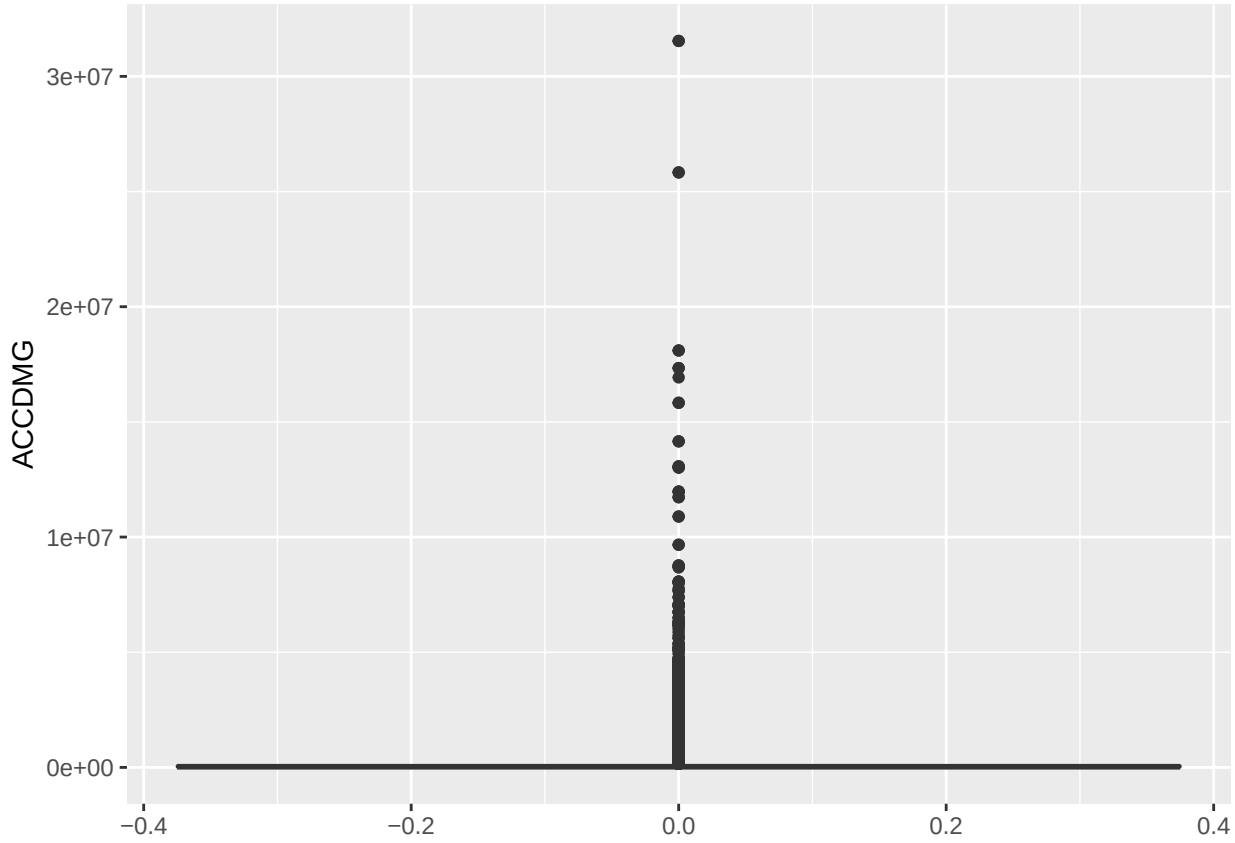

```
tbl
```

```
##                                        Var1 Freq
## 1        (E) Mechanical and Electrical Failures   88
## 2           (H) Train operation - Human Factors  734
## 3 (M) Miscellaneous Causes Not Otherwise Listed 2397
## 4                   (S) Signal and Communication   13
## 5              (T) Rack, Roadbed and Structures  185
```

Miscellaneous causes was by far the most frequent accident type followed by human factors.

Looking at extreme accidents

```
dmgbox <- ggplot(totacts, aes(y=ACCDMG)) + geom_boxplot()
dmgbox
```




```
# Names associated with box plot features:
names(ggplot_build(dmgbox)$data[[1]])
```

```
##  [1] "ymin"        "lower"       "middle"      "upper"       "ymax"
##  [6] "outliers"    "notchupper"  "notchlower"  "x"           "flipped_aes"
## [11] "PANEL"       "group"       "ymin_final"  "ymax_final"  "xmin"
## [16] "xmax"        "xid"         "newx"        "new_width"   "weight"
## [21] "colour"      "fill"        "size"        "alpha"       "shape"
## [26] "linetype"
```

```
# ymax is the upper whisker - anything above that is an outlier
upper <- ggplot_build(dmgbox)$data[[1]]$ymax

# create a new data frame with only the outliers
xdmg <- totacts[totacts$ACCDMG > upper,]

# how many outliers are there
nrow(xdmg)
```

```
## [1] 7888
```

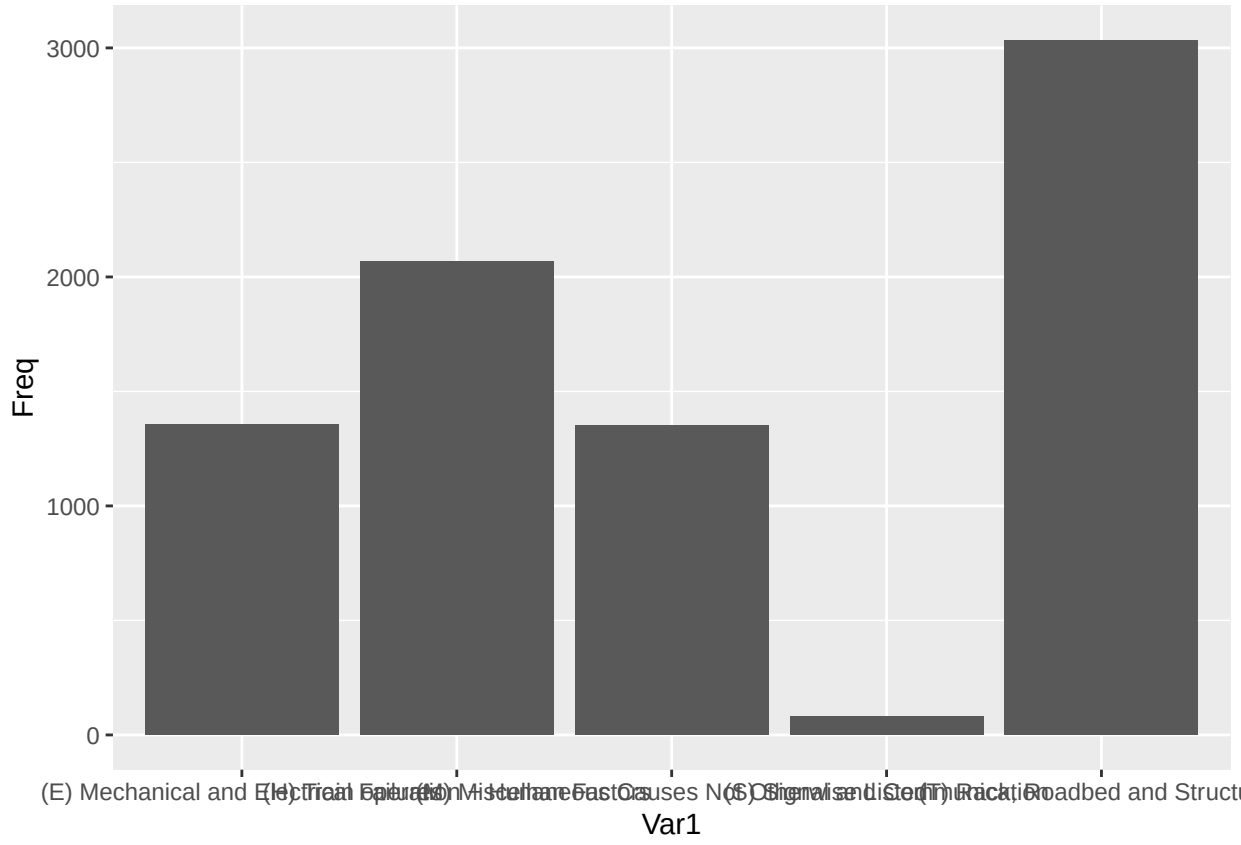
```
# What proportion of accidents are extreme?

frac_acts_x <- round(nrow(xdmg)/nrow(totacts), 2)*100
```

```
# Proportion of costs

frac_cost_x <- round(sum(as.numeric(totacts$ACCDMG[which(totacts$ACCDMG > ggplot_build(dmgbox)$data[[1]]
```

There are 7888 outliers which comprise 75% of the sum of accident damage across all accidents. 13% of all accidents are extreme accidents.

Make a barplot of the cause of accidents for extreme accidents.

```
# Use table() to see the frequencies

tbl <- as.data.frame(table(xdmg$Cause))

# Use barplot() to graph this

ggplot(tbl, aes(x = Var1, y= Freq)) + geom_bar(stat="identity")
```



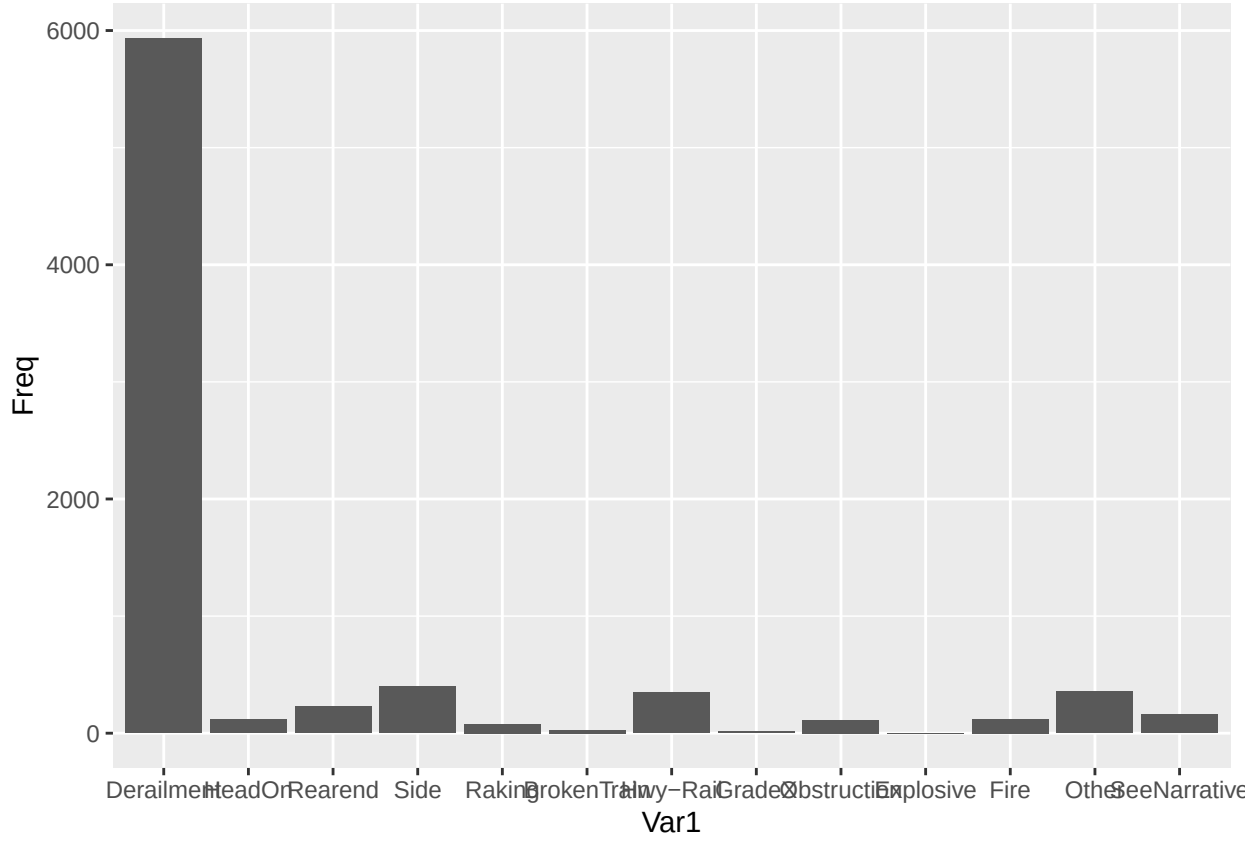

```
tbl
```

```
##                                      Var1 Freq
## 1  (E) Mechanical and Electrical Failures 1356
## 2     (H) Train operation - Human Factors 2067
```

```
## 3 (M) Miscellaneous Causes Not Otherwise Listed 1352
## 4                   (S) Signal and Communication   79
## 5              (T) Rack, Roadbed and Structures 3034
```

Make a barplot of the cause of accidents for extreme accidents.

```
# Use table() to see the frequencies

tbl <- as.data.frame(table(xdmg$TYPE))

# Use barplot() to graph this

ggplot(tbl, aes(x = Var1, y= Freq)) + geom_bar(stat="identity")
```




```
tbl
```

```
##            Var1 Freq
## 1    Derailment 5933
## 2        HeadOn  115
## 3       Rearend  226
## 4          Side  397
## 5        Raking   78
## 6   BrokenTrain   28
```

```
## 7      Hwy-Rail  346
## 8        GradeX   13
## 9   Obstruction  113
## 10    Explosive    2
## 11         Fire  121
## 12        Other  358
## 13 SeeNarrative  158
```

Box plot of extreme accidents by accident type

ACCDMG Analysis:

Casualties Analysis: