Global Terrorism DB Project

Evan Harley

April 13, 2015

First Steps

The very first things that I did were load the appropriate libraries and the database that I would be working with.

```
setwd('E:/Downloads/GTD_0814dist')
library(ggplot2)
library(openxlsx)
library(dplyr)
data <- read.xlsx('globalterrorismdb_0814dist.xlsx',
                  colNames = TRUE, detectDates = TRUE)
```

Data Wrangling

So the first thing that I noticed about this data set is the fact that there are rather a lot of variables in the data set that repeat the same information in. So, I decided that I would work with a subset of the variables. I chose the variables that made the most sense to me.

```
#Selecting the Subset of the variables that I will be using
keeps <- c('eventid', 'iyear', 'imonth', 'iday', 'extended',
           'resolution', 'country', 'country_txt', 'region',
           'region_txt', 'city',  'doubtterr', 'multiple',
           'success', 'suicide', 'attacktype1', 'attacktype1_txt',
           'targtype1', 'targtype1_txt', 'weaptype1', 'weaptype1_txt',
           'gname', 'nperps',  'nkill','nkillter',
           'nwound', 'nwoundte', 'property', 'propextent')
```

I chose to only keep the first attack type, target type, and weapon type, because the majority of the entries that I looked at did not have values. I chose to drop many of the kidnapping/hostage/hijacking specific values because they applied to a subset of the values that I wasn't particularly interested in.

After taking a look at the values in the Weapon Type 1 text variable I noticed that the vehicle variable value was long enough to obscure the value of a count, so I subset the data and changed the value to just Vehicle

```
# Setting 2 invalid weapon types so that I can later take a long string and make it one # word
invalid_weap <- "Vehicle (not to include vehicle-borne explosives, i.e., car or truck bombs)"
invalid_weap2 <- "Explosives/Bombs/Dynamite"
data$weaptype1_txt[data$weaptype1_txt == invalid_weap] <- "Vehicle"
data$weaptype1_txt[data$weaptype1_txt == invalid_weap2] <- "Explosives"
```

Just to ensure that there aren't any date values that don't make sense I ran a tally of all of the day and month variables

```
tally(group_by(data, imonth))
```

```
## Source: local data frame [13 x 2]
##
##    imonth     n
## 1       0    23
## 2       1  9994
## 3       2  9152
## 4       3 10453
## 5       4 10401
## 6       5 11451
## 7       6 10541
## 8       7 11127
## 9       8 11005
## 10      9  9822
## 11     10 10999
## 12     11 10565
## 13     12  9554
```

```r
tally(group_by(data, iday))
```

```
## Source: local data frame [33 x 2]
##
##    iday    n
## 1   -99    1
## 2     0  895
## 3     1 4499
## 4     2 4173
## 5     3 4089
## 6     4 4276
## 7     5 3995
## 8     6 3970
## 9     7 4163
## 10    8 3960
## ..  ...  ...
```

```r
#removing invalid date data
invalid_date <- data[data$iday == -99, ]
data <- data[data$imonth != 0, ]
data <- data[data$iday != 0, ]
data <- data[data$iday != -99, ]
```

Finding some wrangling neeed I continued to tally variables looking for values not accounted for in the codebook document.

```r
tally(group_by(data, extended))
tally(group_by(data, country))
tally(group_by(data, region))
tally(group_by(data, doubtterr))
tally(group_by(data, multiple))
tally(group_by(data, success))
tally(group_by(data, suicide))
tally(group_by(data, attacktype1))
tally(group_by(data, gname))
tally(group_by(data, nperps))
tally(group_by(data, nkill))
tally(group_by(data, nkillter))
tally(group_by(data, nwound))
tally(group_by(data, nwoundte))
tally(group_by(data, property))
tally(group_by(data, propextent))

# Taking the "NA" values in number wounded and killed variables and setting them to zero as the code bo

data$nkill[is.na(data$nkill)] <- 0
data$nkillter[is.na(data$nkillter)] <- 0
data$nwound[is.na(data$nwound)] <- 0
data$nwoundte[is.na(data$nwoundte)] <- 0
```

The only values that took me aback are the values between whole numbers in the nkill, nkillter, nwound, and nwoundte variables. These are explained in the literature as averages.

I also noticed that there was no consistant date variable. So, I added one

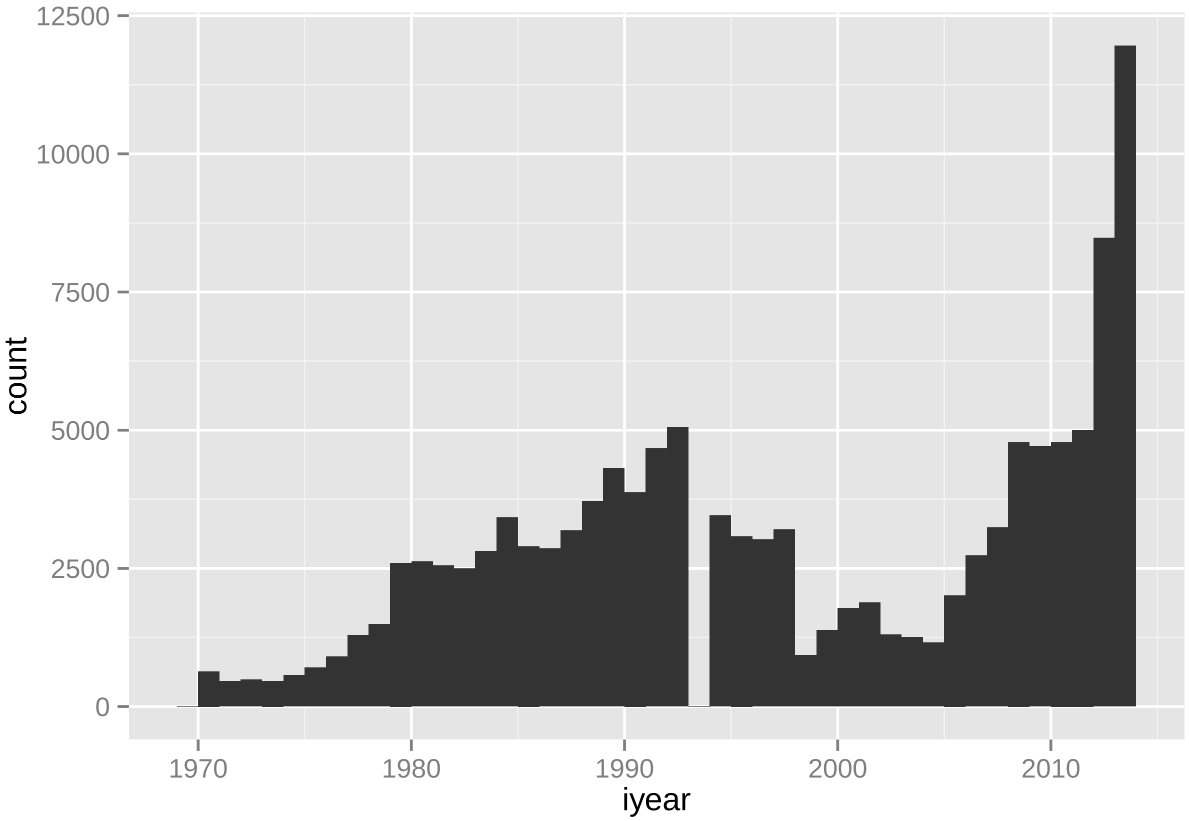
```
data$date <- paste(data$iyear, data$imonth, data$iday, sep = "/")
data$date = as.Date(data$date)
data$day_of_week <- weekdays(data$date)
```

Single Variable Explorations

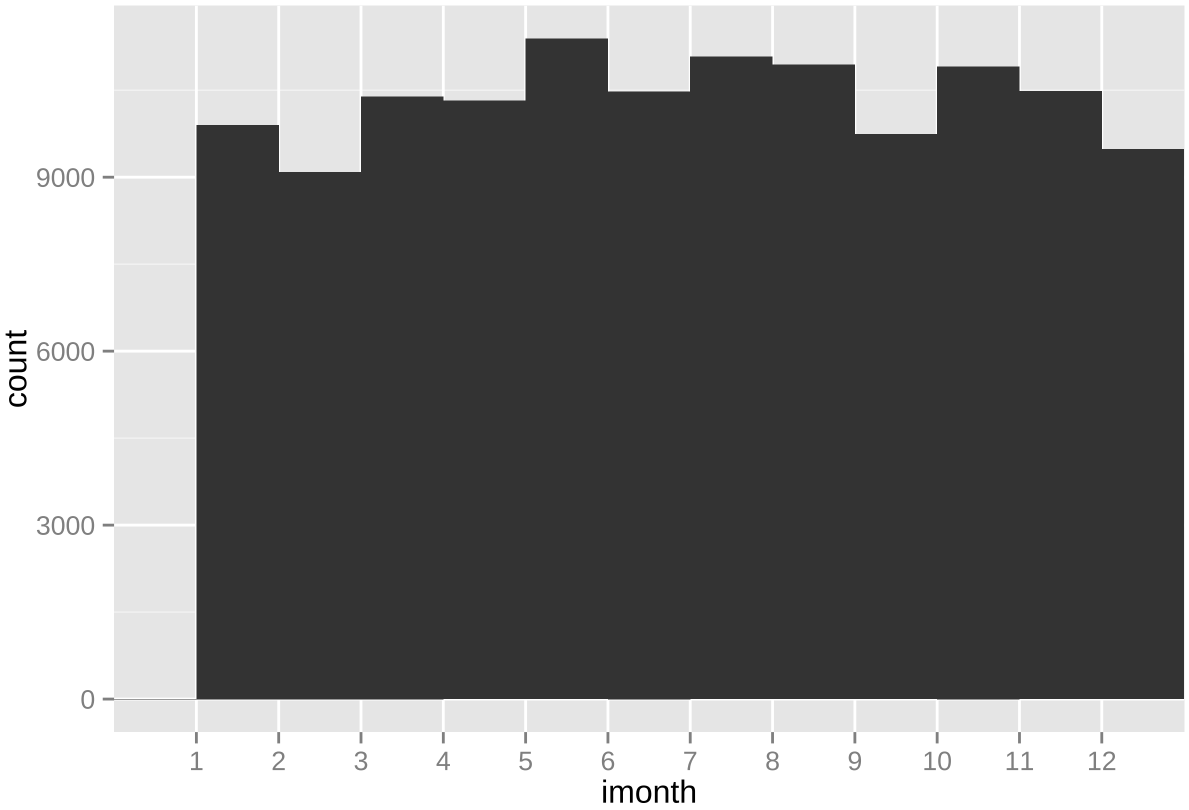
```
summary(data$iyear)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1970    1990    2000    2000    2010    2010
```

```
ggplot(aes(x=iyear), data = data) +
  geom_histogram(binwidth = 1)
```



```
ggplot(aes(x=imonth), data = data) +
  geom_histogram(binwidth = 1)+
  scale_x_discrete(limits= 1:12)
```



The first plot that I plotted was a histogram of years which shows an almost linear increase in incidence of terrorist attacks up until 1992 where there is a gap in the data set. This is due to a loss of data in the data set, which resulted in a total number of incidents that only totaled 15% of the previous estimate of incident numbers. While there is not specific data there is an estimate in the explanatory document of 4954 incidents in that year. This would make sense given the decrease in number of incidents in 1994. There is a significantly lower number of incidents between 1998 and 2004, before spiking again in what looks like an exponential increase.

There doesn't seem to be much variance in the month variable.

```
tally(group_by(data, iday))
```

```
## Source: local data frame [31 x 2]
##
##    iday    n
## 1     1 4499
## 2     2 4173
## 3     3 4089
## 4     4 4276
## 5     5 3995
## 6     6 3970
## 7     7 4163
## 8     8 3960
## 9     9 4238
## 10   10 4206
## ..  ...  ...
```

When I looked at the tally for days, the distribution seems pretty flat.

```
arrange(tally(group_by(data, region_txt)), desc(n))
```

```
## Source: local data frame [13 x 2]
##
##                                       region_txt     n
## 1                                    South Asia 28170
## 2                    Middle East & North Africa 27506
## 3                                 South America 18032
## 4                                Western Europe 15160
## 5                   Central America & Caribbean 10490
## 6                            Sub-Saharan Africa  9068
## 7                                Southeast Asia  8188
## 8                                 North America  2887
## 9  Russia & the Newly Independent States (NIS)  2481
## 10                               Eastern Europe  1028
## 11                                   East Asia   704
## 12                                Central Asia   248
## 13                        Australasia & Oceania   229
```

The next table is of the region variable. This tally shows a low number in North America, East Asia, Central Asia, Eastern Europe, Russia, and Australasia. With high numbers in Central America, South America, Western Europe, and the Middle East.

```
arrange(tally(group_by(data, attacktype1_txt)), desc(n))
```

```
## Source: local data frame [9 x 2]
##
##                        attacktype1_txt     n
## 1                    Bombing/Explosion 59233
## 2                        Armed Assault 29979
## 3                        Assassination 15617
## 4      Facility/Infrastructure Attack  7339
## 5         Hostage Taking (Kidnapping)  6371
## 6                              Unknown  3813
## 7 Hostage Taking (Barricade Incident)   686
## 8                      Unarmed Assault   685
## 9                            Hijacking   468
```

This table shows that the three most commonly reported terorist incidents are Assassinations, Armed Assaults, and Bombings. It aslo shows that the least common terrorist actions are Barricade Incidents, Hijackings, and Unarmed Assaults.

```
arrange(tally(group_by(data, targtype1_txt)), desc(n))
```

```
## Source: local data frame [22 x 2]
##
##                      targtype1_txt     n
## 1     Private Citizens & Property 27220
## 2            Government (General) 16653
## 3                        Business 16437
## 4                          Police 16418
## 5                        Military 16267
## 6                  Transportation  5682
## 7                       Utilities  4916
## 8         Educational Institution  3287
## 9  Religious Figures/Institutions  3005
## 10        Government (Diplomatic)  2977
## ..                            ...   ...
```

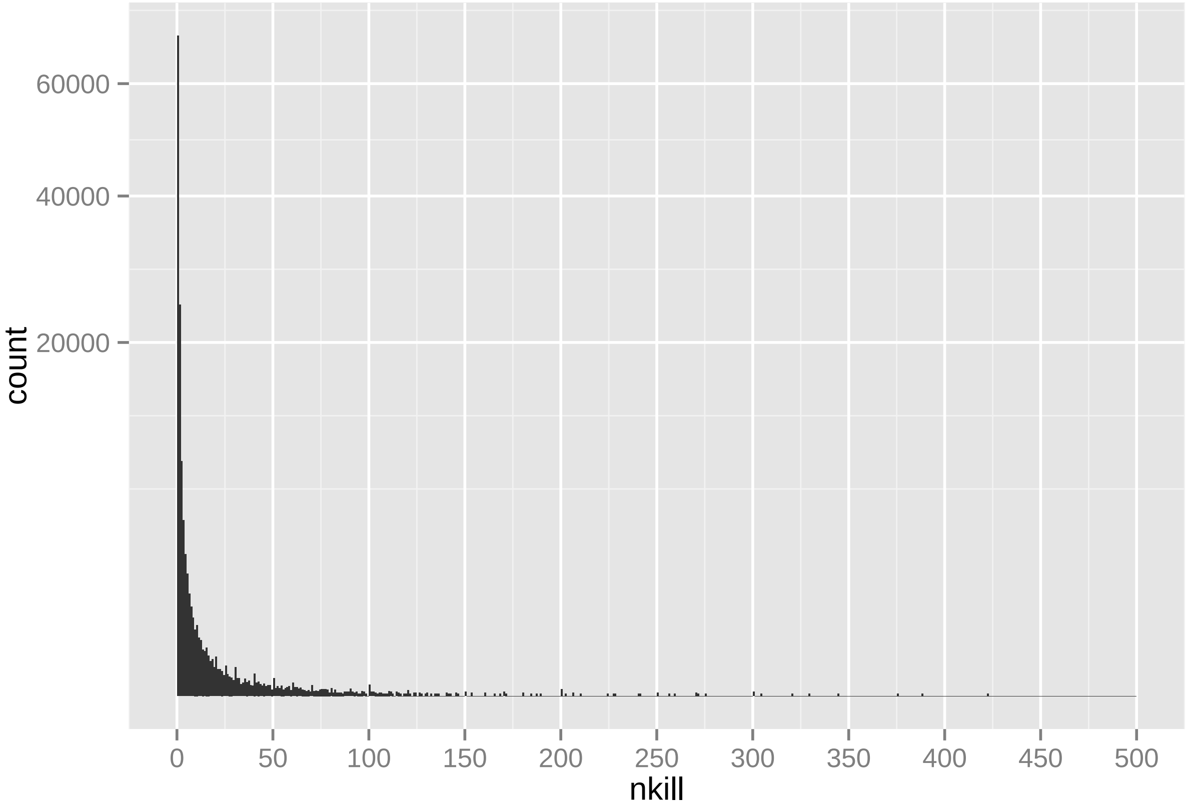
The table is displaying the distribution of target types shows that there are 5 most common target types with the most common being Private Citizens, and the other four being businesses, government, police, and military buildings.

```
arrange(tally(group_by(data, weaptype1_txt)), desc(n))
```

```
## Source: local data frame [12 x 2]
##
##         weaptype1_txt     n
## 1          Explosives 60825
## 2            Firearms 42721
## 3             Unknown  9301
## 4          Incendiary  8442
## 5               Melee  2385
## 6            Chemical   205
## 7  Sabotage Equipment   110
## 8               Other    68
## 9             Vehicle    57
## 10         Biological    33
## 11       Fake Weapons    31
## 12       Radiological    13
```

Firearms and explosives are the the most common weapon types.

```
ggplot(aes(x = nkill), data = data) +
  geom_histogram(binwidth = 1) +
  scale_y_sqrt() +
  scale_x_continuous(limits = c(0, 500), breaks = seq(0,500,50))
```

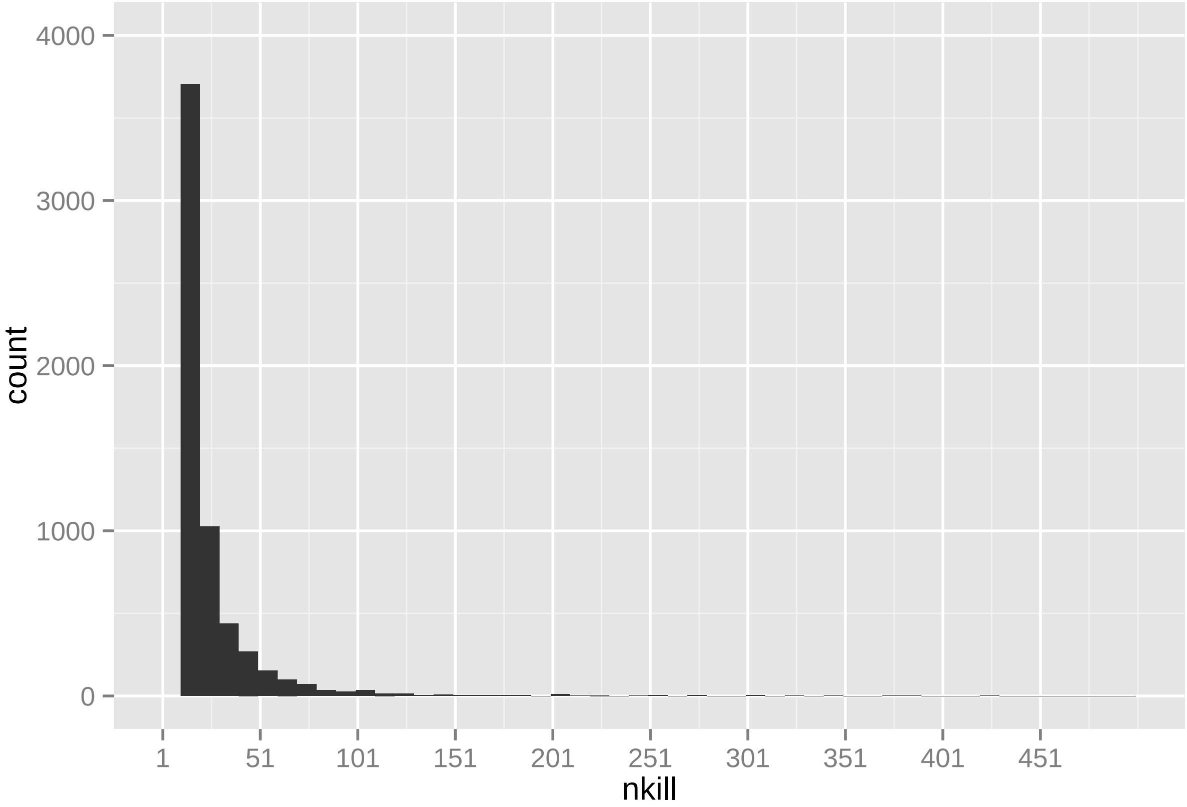


So with this histogram I noticed that there was a long tail, and that the number of attacks with less than ten kiled was a very large number. So, I made another histogram looking at less than 500 killed, and set the binwidth to 1 so that I could see what the most common number killed was. It turned out to be 0. So below find a histogram of number killed where number killed is greater than 0

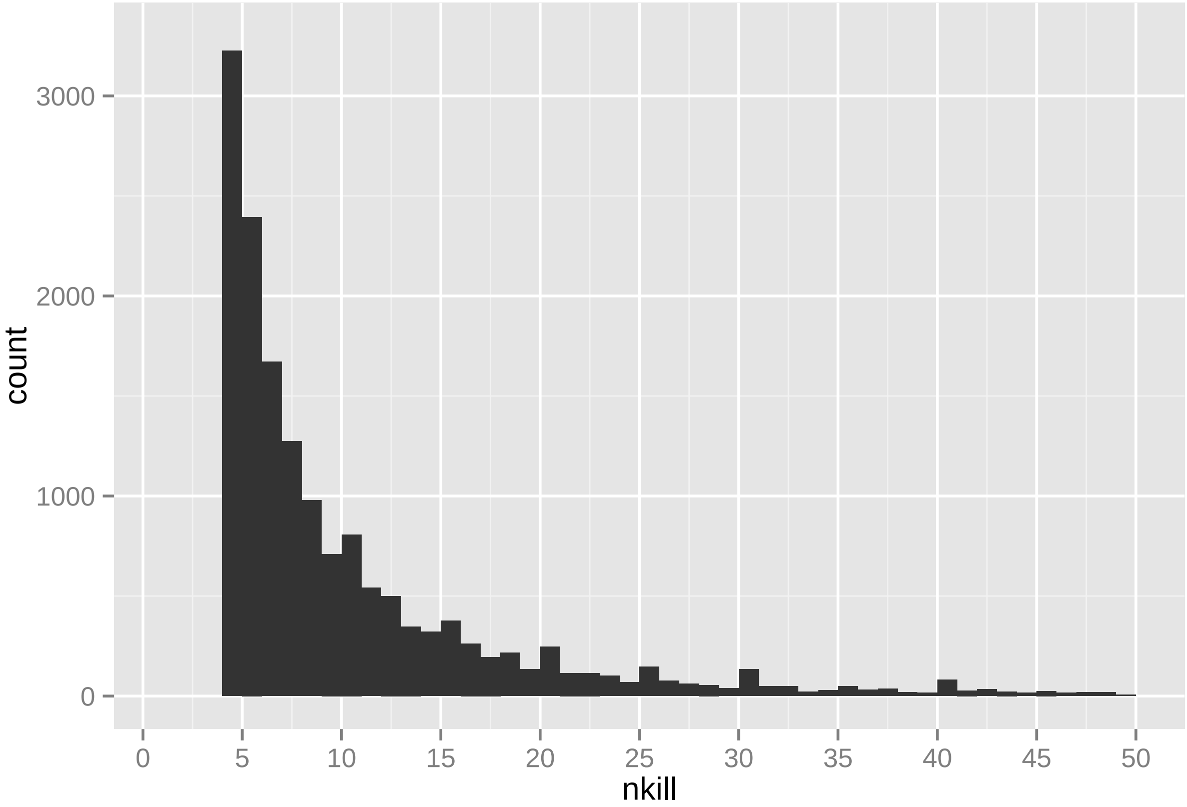
```
summary(data$nkill)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     0.0     0.0     2.1     1.0  1380.0
```

```
ggplot(aes(x = nkill), data = data)+
  geom_histogram(binwidth = 10)+
  scale_x_continuous(limits = c(1, 500), breaks = seq(1, 500, 50))+
  scale_y_continuous(limits = c(0, 4000))
```



```r
ggplot(aes(x = nkill), data = subset(data, data$nkill > 0))+
  geom_histogram(binwidth = 1)+
  scale_x_continuous(limits = c(1, 50), breaks = seq(0, 50, 5))+
  scale_y_continuous(limits = c(0, 3300))
```



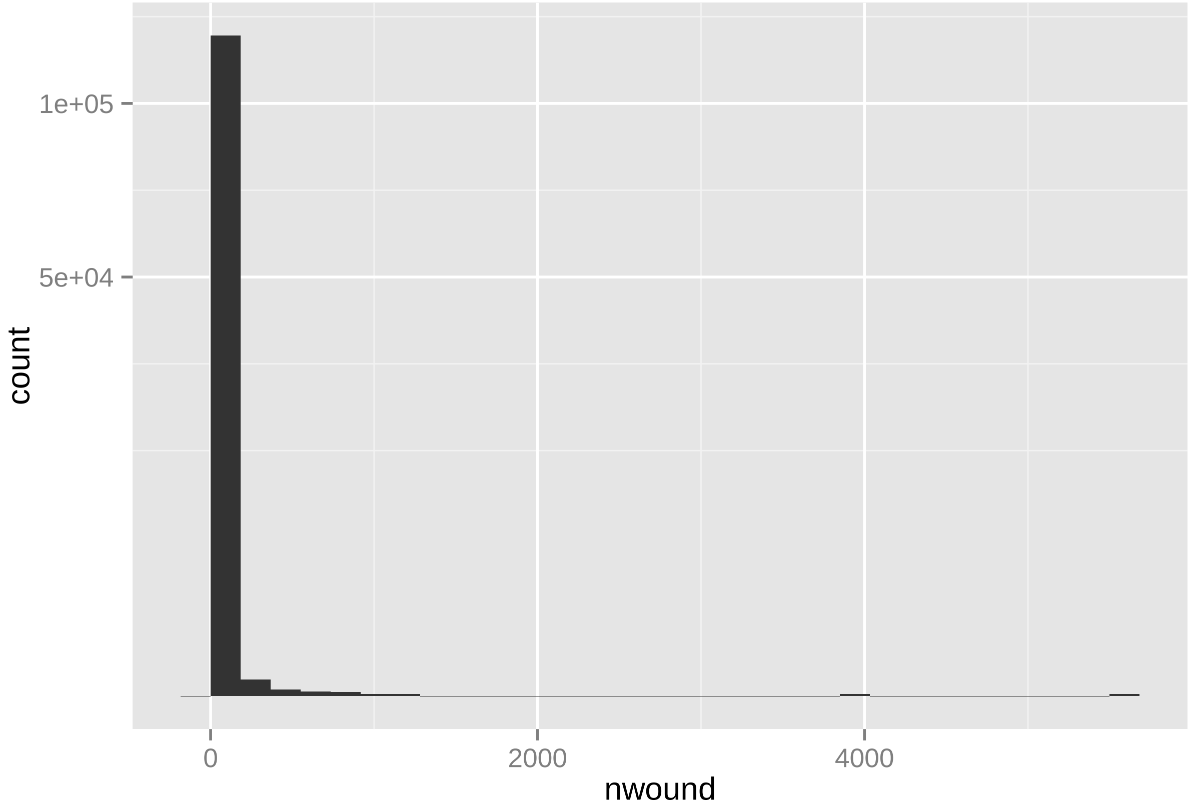
The above histograms just display the distribution of kills both including 0 kill incidents and excluding them.

```
summary(data$nwound)
```

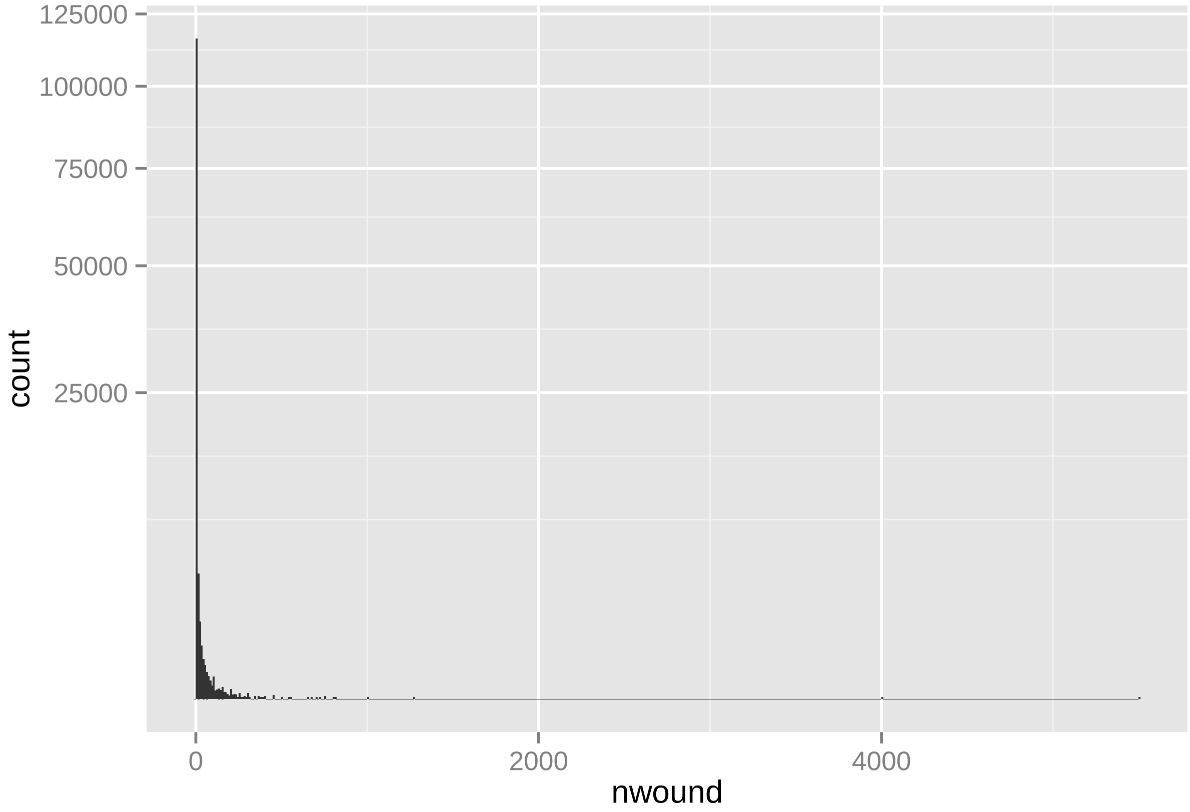
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0       0       0       3       1    5500
```

```
ggplot(aes(x = nwound), data = subset(data, !is.na(data$nwound)))+
  geom_histogram()+
  scale_y_sqrt()
```

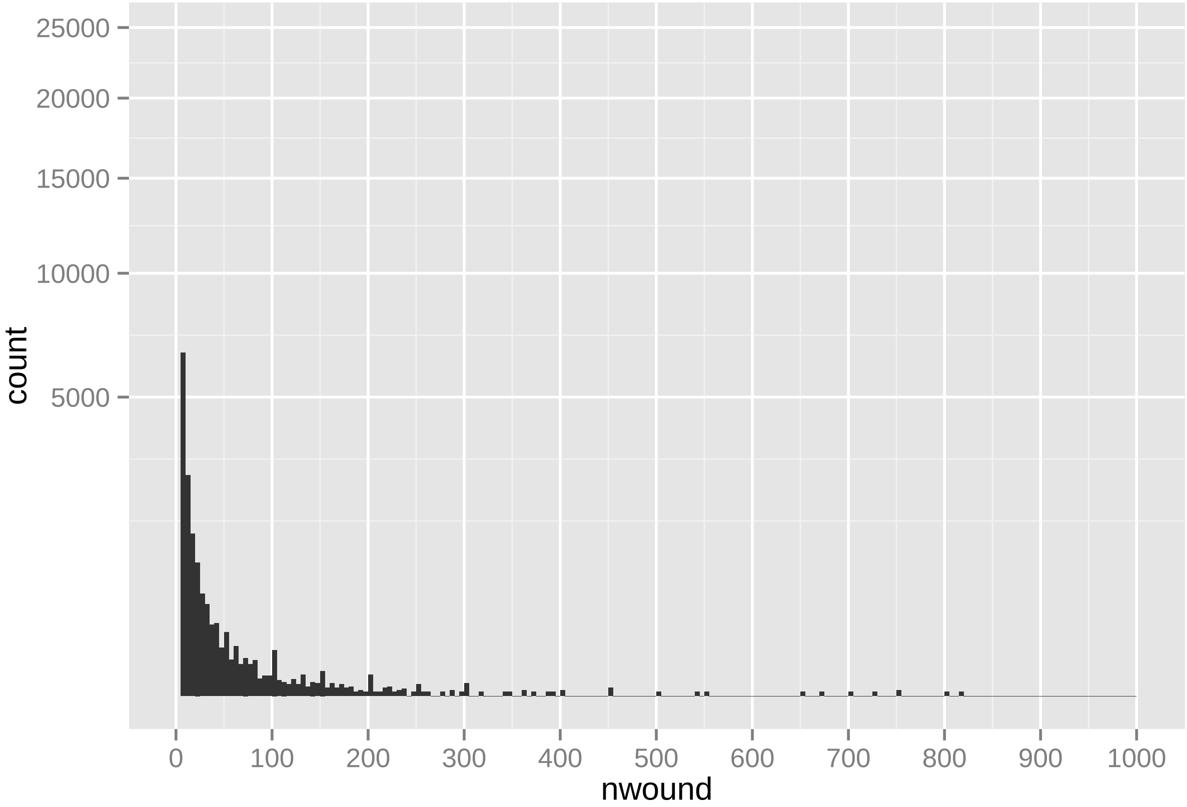
```
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
```



```
ggplot(aes(x = nwound), data = subset(data, !is.na(data$nwound)))+
  geom_histogram(binwidth = 10)+
  scale_y_sqrt()
```



```
ggplot(aes(x = nwound), data = subset(data, !is.na(data$nwound)))+
  geom_histogram(binwidth = 5)+
  scale_y_sqrt()+
  scale_x_continuous(limits = c(1, 1000), breaks = seq(0, 1000, 100))
```



The above shows a snapshot of the steps that I used to build the final histogram showing the distribution of number of people wounded in terrorist incidents. It shows that, as above with the number killed variable, the majority of incidents have fewer than 0 people wounded.

```
tally(group_by(data, country_txt))
```

```
## Source: local data frame [209 x 2]
##
##            country_txt    n
## 1          Afghanistan 5931
## 2              Albania   71
## 3              Algeria 2660
## 4              Andorra    1
## 5               Angola  474
## 6  Antigua and Barbuda    2
## 7            Argentina  797
## 8              Armenia   20
## 9            Australia   73
## 10             Austria  105
## ..                 ...  ...
```

```
tally(group_by(data, city))
```

```
## Source: local data frame [27,859 x 2]
##
```

```
##                                     city n
## 1                    'A'Ishah Bakkar 1
## 2                             'Alayh 1
## 3                     'Ayta al-Sha'b 1
## 4      (Finca Los Manzanos) Las Palmas 1
## 5   (Lesotho) at Harakolo in Kulinyama 1
## 6                  (Mar Tagla) Beirut 1
## 7                                    * 6
## 8                 10 mi S. of Kampala 1
## 9                   15 September Dam 1
## 10    15 km from South African border 1
## ..                                 ... .
```

This showed me some interesting information namely that there are about 28,000 city values, and that many of the city values actually hold a value describing the location of the incident. So, I will not likely use that feature in my visualizations.

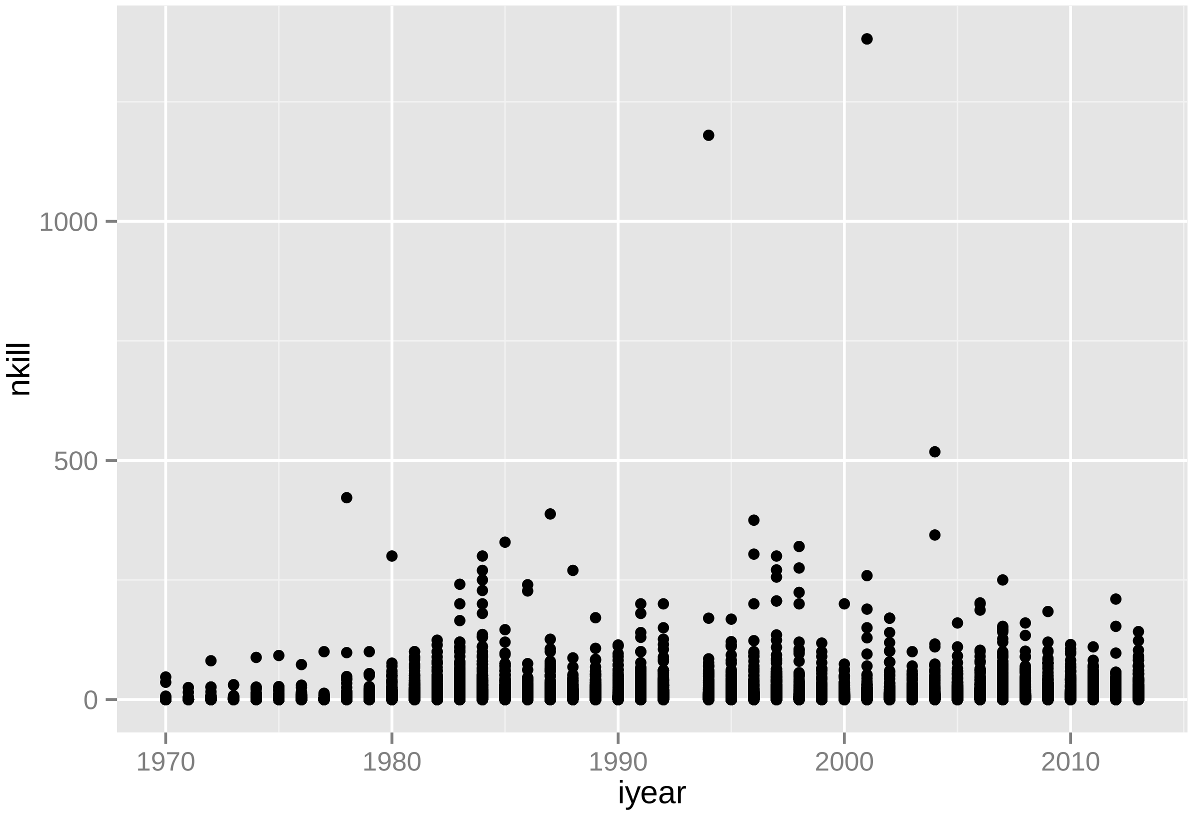
```
tally(group_by(data, day_of_week))
```

```
## Source: local data frame [7 x 2]
##
##   day_of_week     n
## 1      Friday 16584
## 2      Monday 19263
## 3    Saturday 15758
## 4      Sunday 16908
## 5    Thursday 18327
## 6     Tuesday 18540
## 7   Wednesday 18811
```

Two Variable Explorations

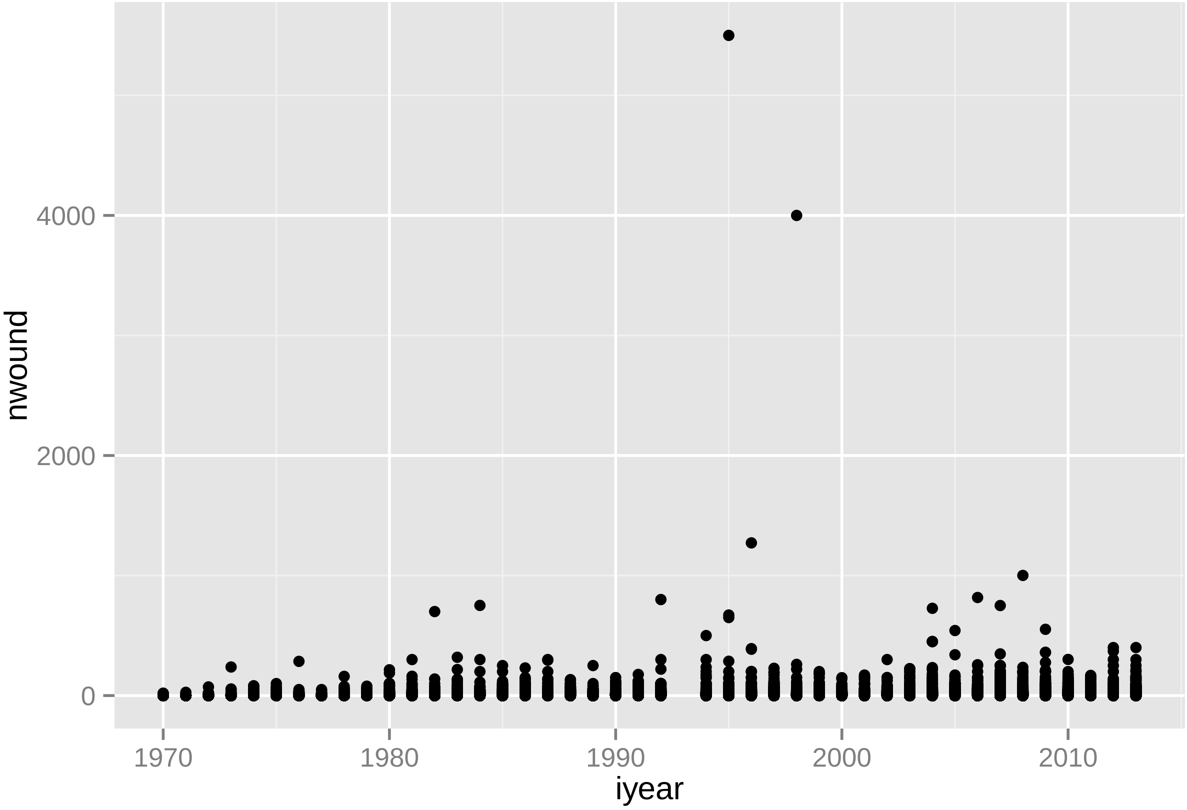
Now I am going to take a look at the relationship between two variables

```
ggplot(aes(x = iyear, y = nkill), data = data)+
  geom_point()
```



The first visualization that I decided to plot was taking a look at the relationship between year and number of people killed in incidents. While there are significant outliers most of the data stays pretty well within a clear distribution. I will likely take a look at this in another multivariate visualization.

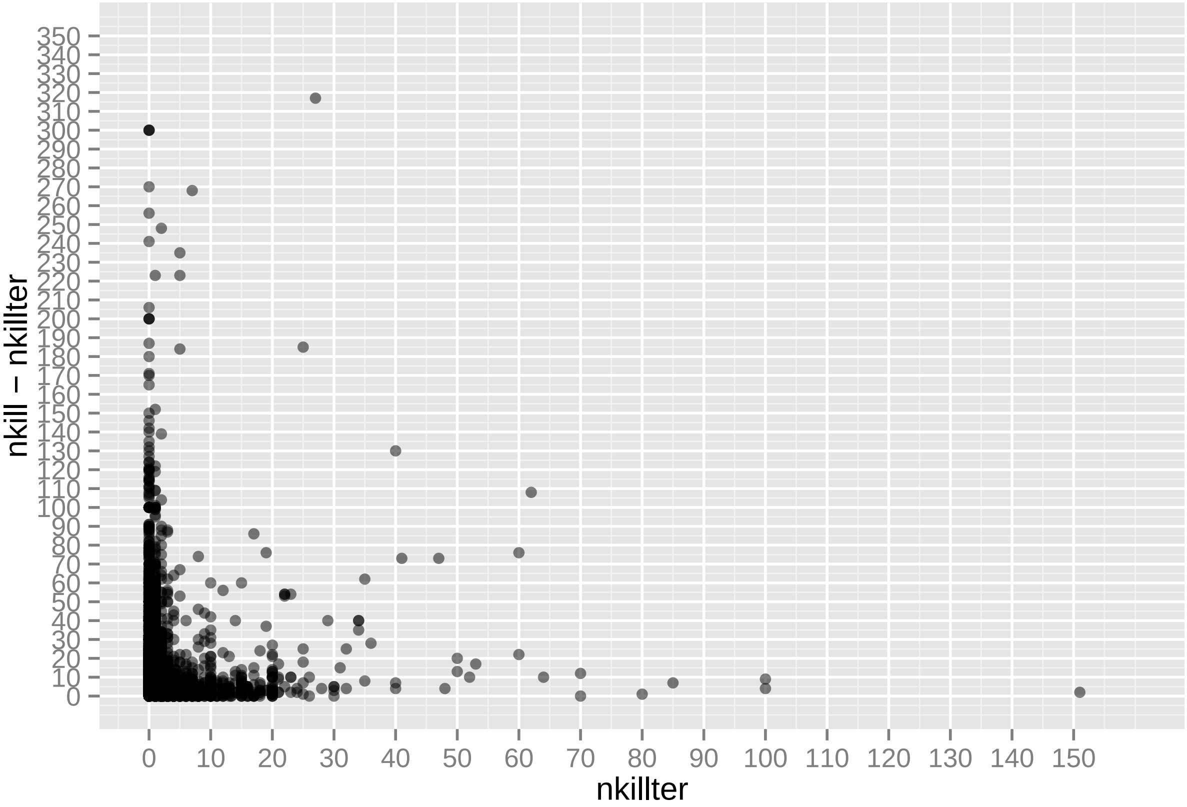
```r
ggplot(aes(x = iyear, y = nwound), data = data)+
  geom_point()
```



I thought it might be interesting to take a look at the number of people injured by year. It is not nearly as interesting in distribution as the number killed distribution

```
ggplot(aes(x = nkillter, y = nkill - nkillter), data = data)+
  geom_jitter(alpha = .50)+
  scale_x_continuous(limits = c(0, 160), breaks = seq(0, 150, 10))+
  scale_y_continuous(limits = c(0, 350), breaks = seq(0, 350, 10))
```

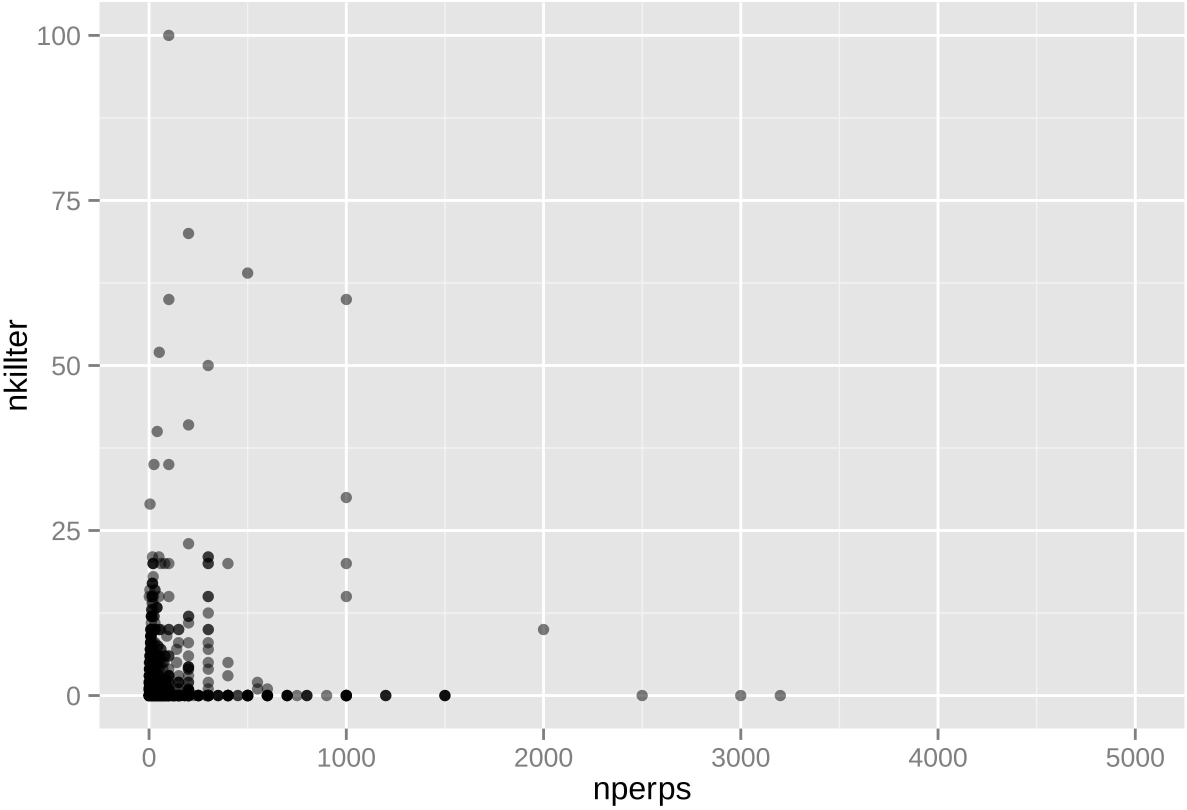
```
## Warning: Removed 77717 rows containing missing values (geom_point).
```



This plot shows a rather sharp clear line where nkillter = 0. I subtracted number of terrorists killed from the total number killed because the number of terrorists killed is included in the total number killed.

```r
ggplot(aes(x = nperps, y = nkillter),
       data = subset(data, !is.na(data$nperps) & data$nperps != -99))+
  geom_jitter(alpha = .5)+
  scale_x_continuous(limits = c(0, 5000))
```

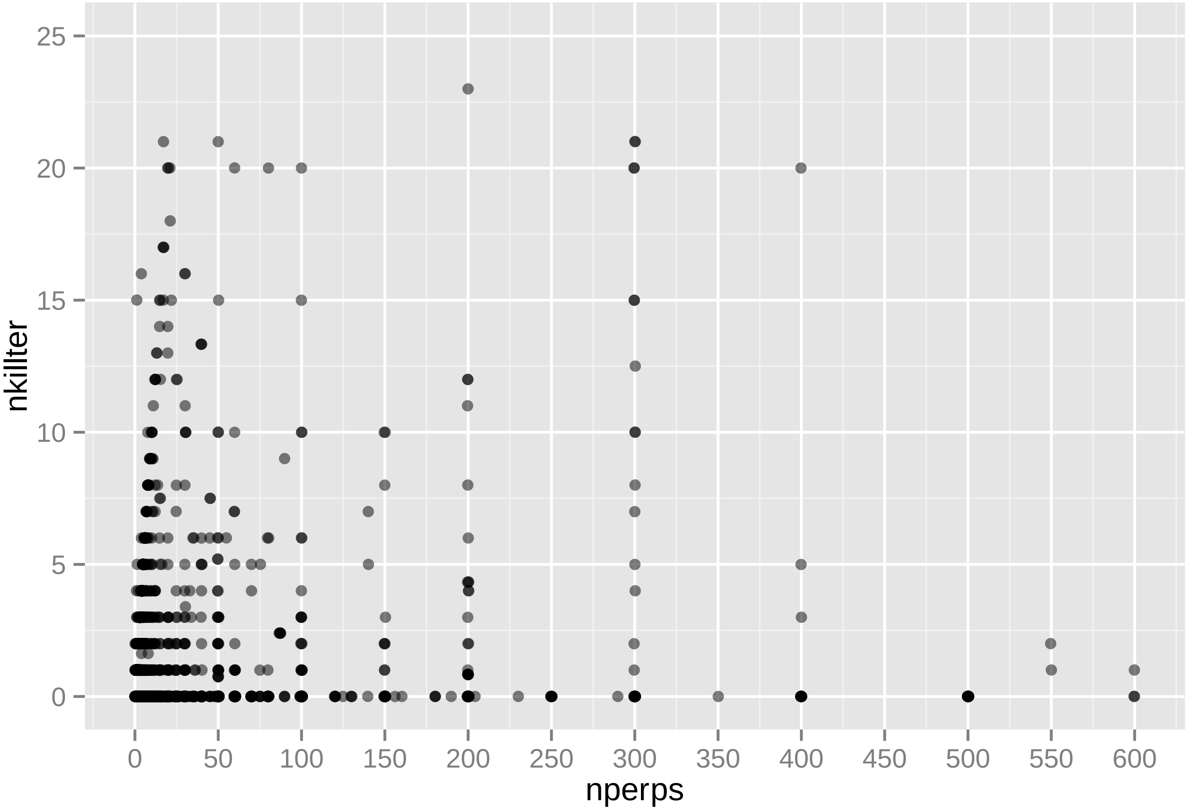
```
## Warning: Removed 104 rows containing missing values (geom_point).
```



This plot compares the total number of perpetrators involved in terrorist acts to the number of terrorists killed. I notice that most of the data points are clustered in the lower left corner, so I plotted the lower left corner of the graph.

```
ggplot(aes(x = nperps, y = nkillter),
       data = subset(data, !is.na(data$nperps) & data$nperps != -99))+
  geom_jitter(alpha = .5)+
  scale_x_continuous(limits = c(0, 600), breaks = seq(0, 600, 50))+
  scale_y_continuous(limits = c(0, 25))
```

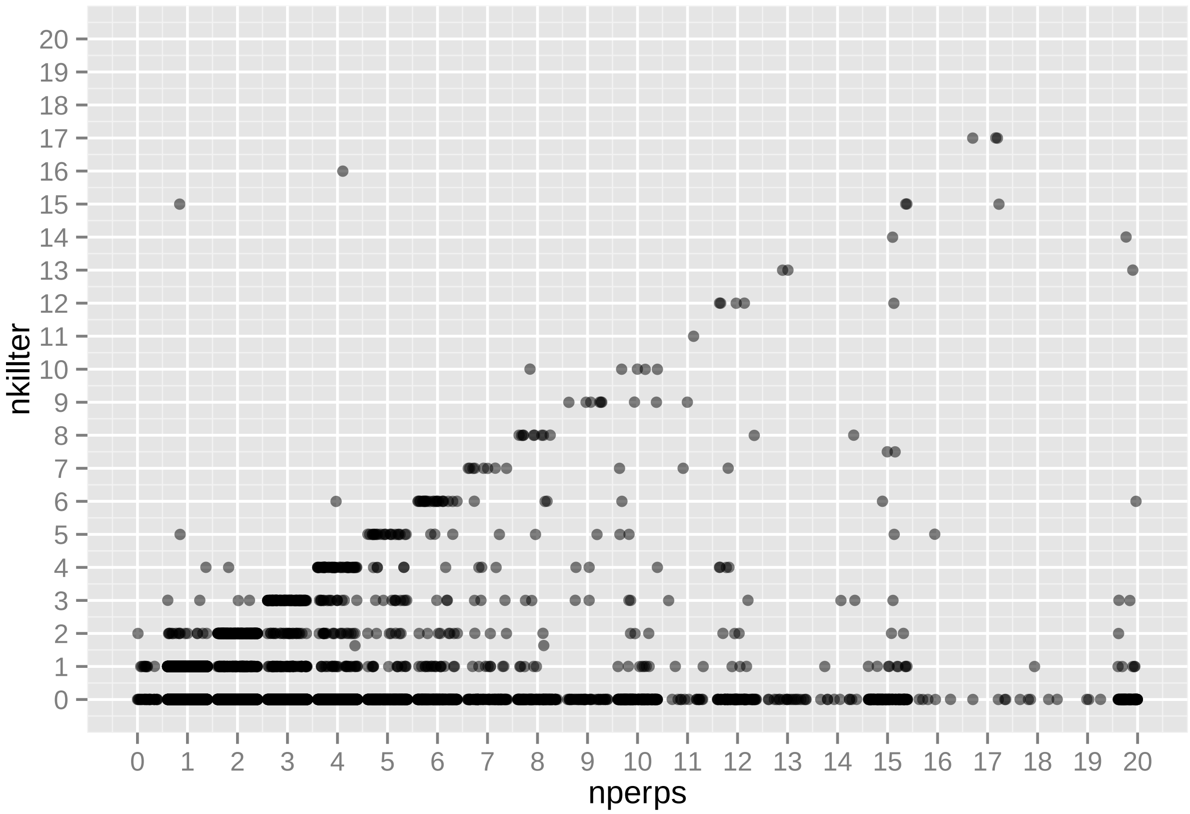
```
## Warning: Removed 7510 rows containing missing values (geom_point).
```



It looks as though numbers over about 50 perps involved are estimaes, as they seem for the most part to fall on numbers divisible by 50. I am going to plot one more plot based on this looking at the lower left corner of the graph yet again.

```
ggplot(aes(x = nperps, y = nkillter),
       data = subset(data, !is.na(data$nperps) & data$nperps != -99))+
  geom_jitter(alpha = .5)+
  scale_x_continuous(limits = c(0, 20), breaks = seq(0, 20, 1))+
  scale_y_continuous(limits = c(0, 20), breaks = seq(0, 20, 1))
```

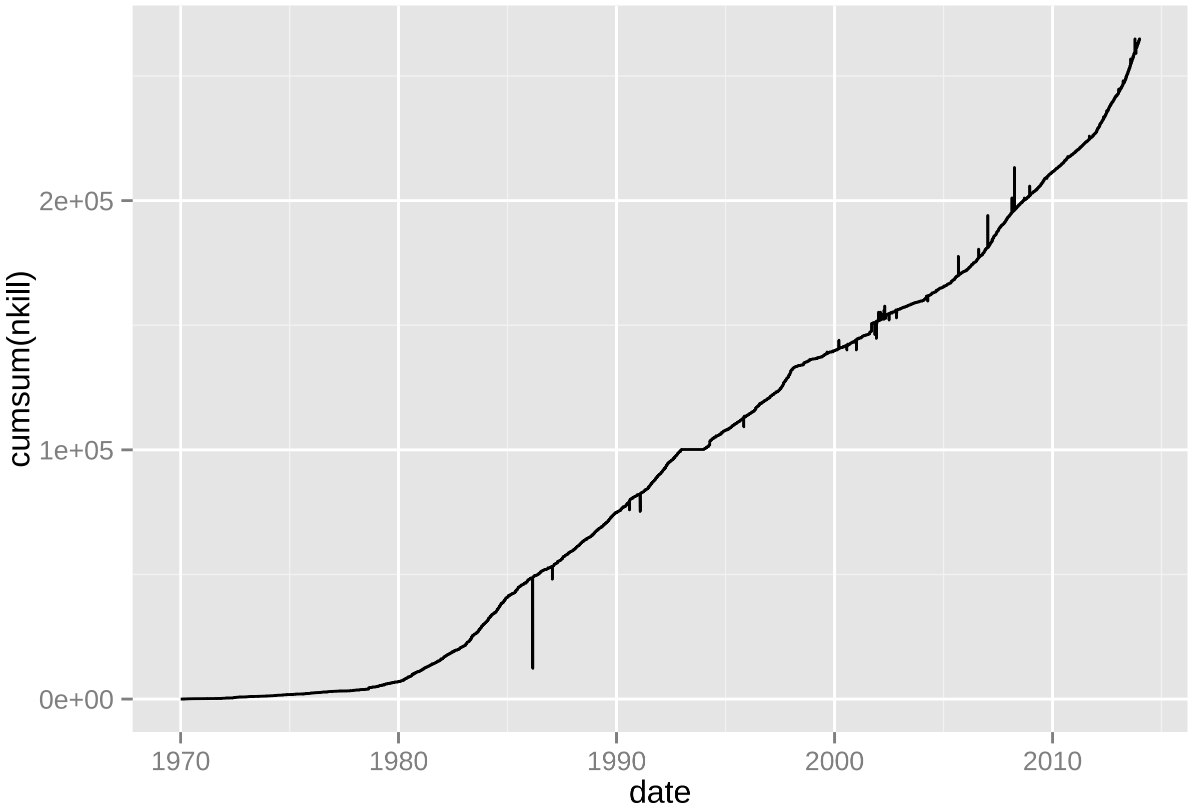
```
## Warning: Removed 9045 rows containing missing values (geom_point).
```



This plot has a horizontal bar at zero that stretches all the way across the plot. This shows that across all number of perpetrators there are many incidents with no resulting deaths to the terrorists involved. Further up the graph we see that with more reported deaths of terorists the more likely it is that it was all of the terrorists that participated. With a few outliers where many more terrorists were killed than participated in the incident.

Then I did a plot plotting the time series of number killed over time to see how sharp the increase is.

```r
ggplot(aes(x = date, y = cumsum(nkill)), data = data)+
  geom_line()
```

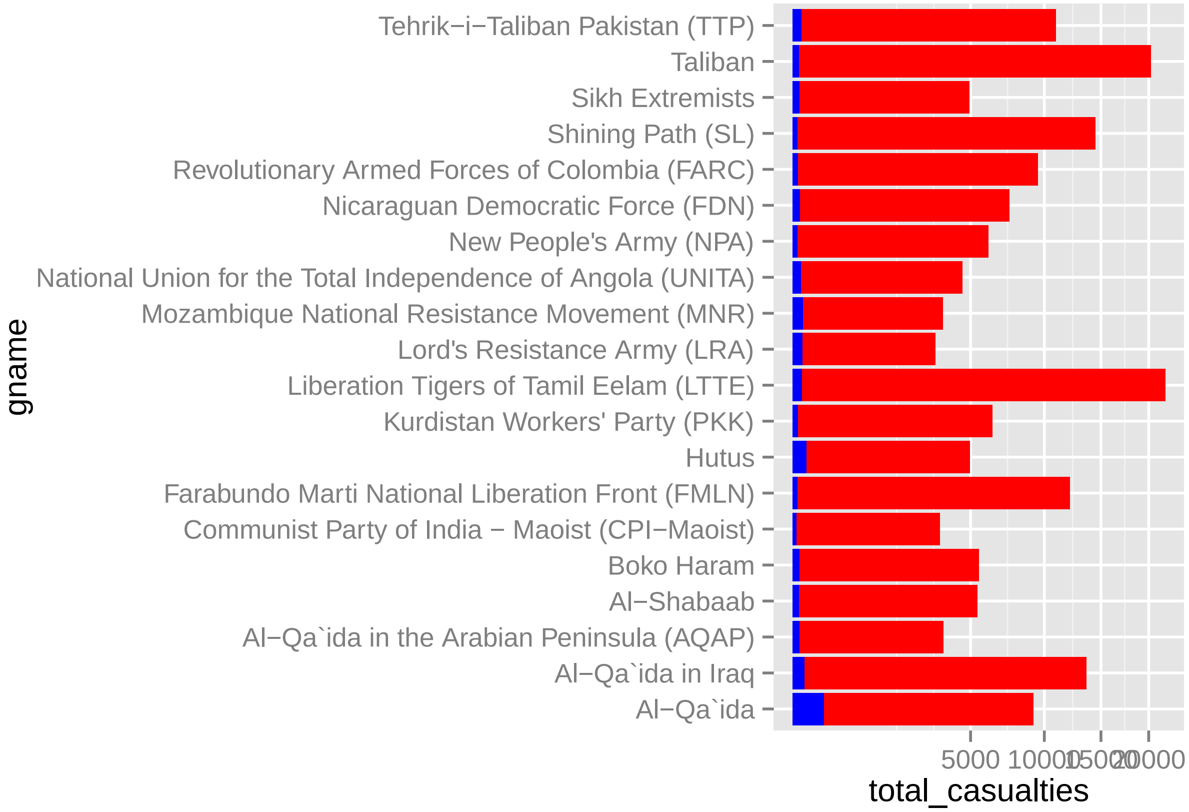


Multivariate Plots

```
data_by_gname <- group_by(subset(data, gname != "Unknown"), gname)
data.nkill_by_group <- summarise(data_by_gname,
                                 mean_killed = mean(nkill),
                                 median_killed = median(nkill),
                                 sum_killed = sum(nkill),
                                 mean_wound = mean(nwound),
                                 median_wound = median(nwound),
                                 sum_wound = sum(nwound),
                                 n = n())
data.nkill_by_group <- arrange(data.nkill_by_group, desc(sum_killed))
nkill_data <- head(data.nkill_by_group, n = 20)
nkill_data$total_casualties <- nkill_data$sum_killed + nkill_data$sum_wound
nkill_data$average_casualties <- nkill_data$total_casualties / nkill_data$n
nkill_data <- arrange(nkill_data, desc(total_casualties))
head(nkill_data)
```

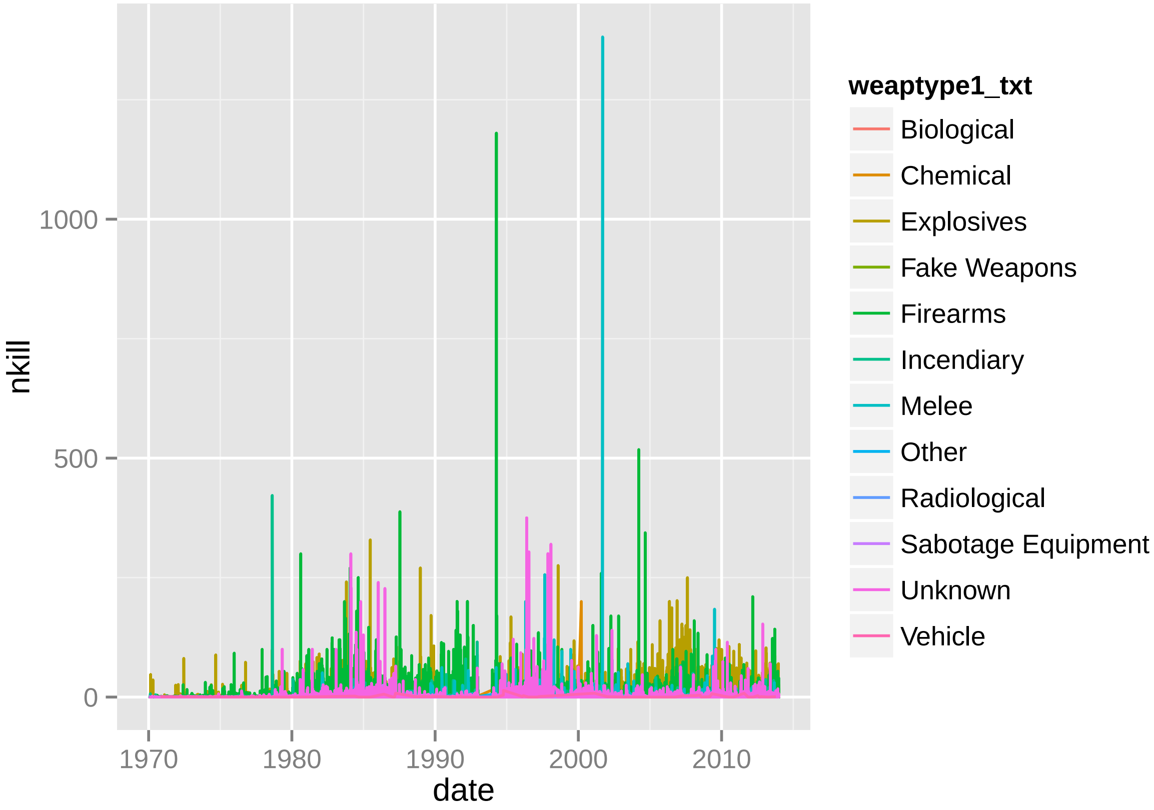
```
## Source: local data frame [6 x 10]
##
##                                              gname mean_killed
## 1          Liberation Tigers of Tamil Eelam (LTTE)       6.838
## 2                                          Taliban       3.154
## 3                                Shining Path (SL)       2.543
## 4                               Al-Qa`ida in Iraq       6.489
## 5 Farabundo Marti National Liberation Front (FMLN)       2.412
## 6                  Tehrik-i-Taliban Pakistan (TTP)       4.760
## Variables not shown: median_killed (dbl), sum_killed (dbl), mean_wound
##   (dbl), median_wound (dbl), sum_wound (dbl), n (int), total_casualties
##   (dbl), average_casualties (dbl)
```

```
ggplot(aes(x = gname, y = total_casualties), data = nkill_data)+
  geom_bar(stat = "identity", fill = "red")+
  geom_bar(aes(y =average_casualties), stat = "identity", fill = "blue")+
  coord_flip()+
  scale_y_sqrt()
```



This is a visualization showing the top 20 terrorist groups as shown by the number of casualties. I find it interesting that a goup that I have never heard of has the most casualties.

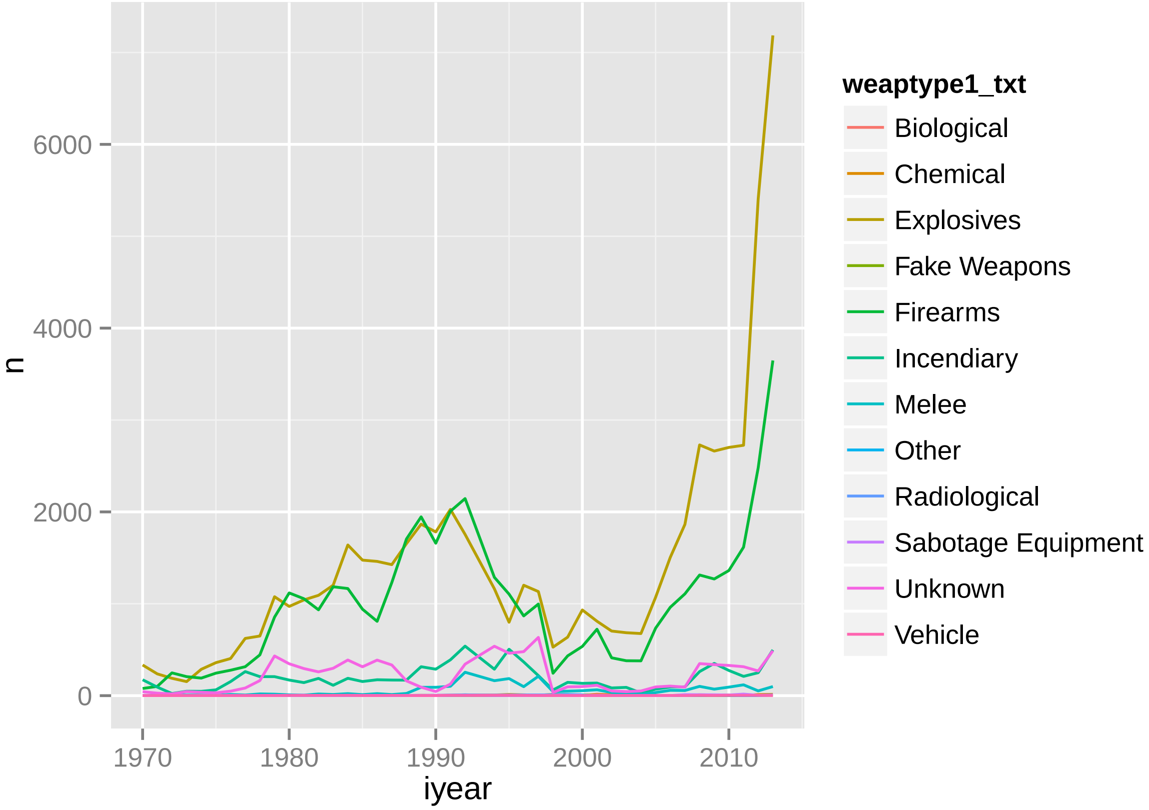
```
ggplot(aes(x = date, y = nkill), data = data)+
  geom_line(aes(color = weaptype1_txt))
```



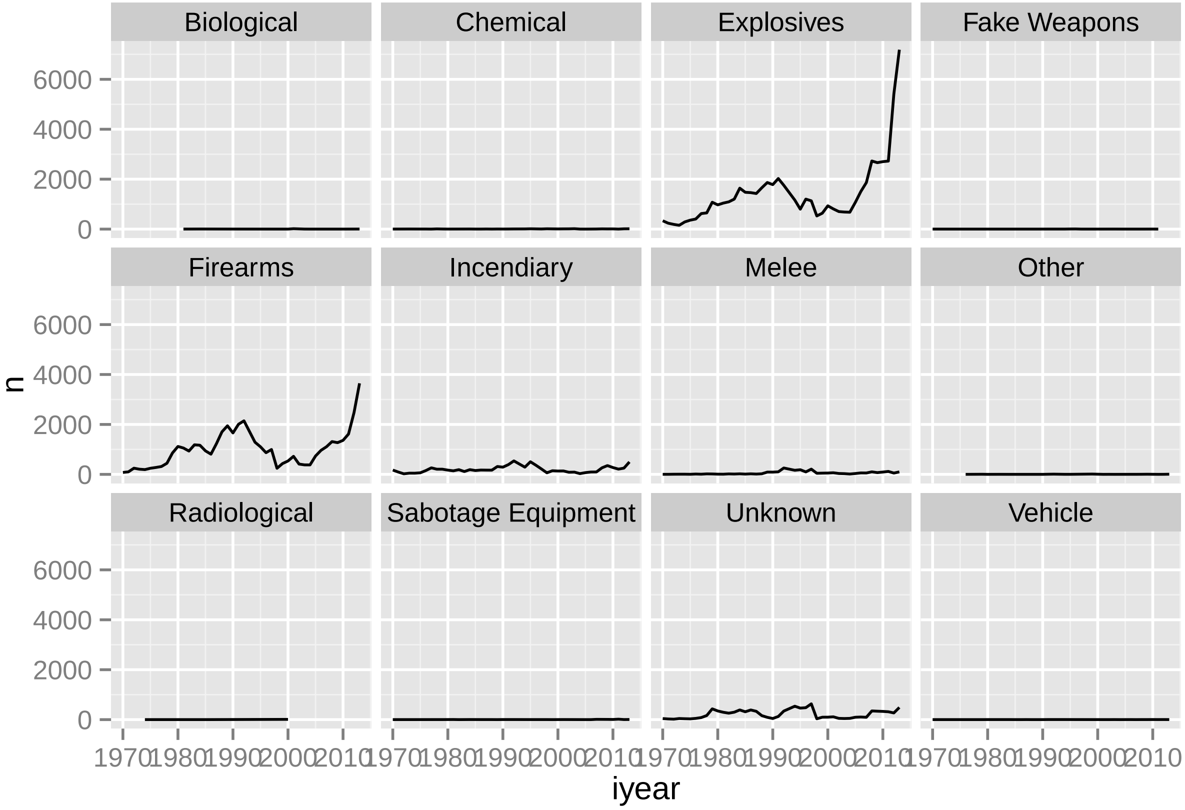
```
data_by_date <- group_by(data, iyear, weaptype1_txt)
by_date_weap <- summarise(data_by_date, n = n())

tally(group_by(data, weaptype1_txt))
```

```
## Source: local data frame [12 x 2]
##
##         weaptype1_txt     n
## 1          Biological    33
## 2            Chemical   205
## 3          Explosives 60825
## 4        Fake Weapons    31
## 5            Firearms 42721
## 6          Incendiary  8442
## 7               Melee  2385
## 8               Other    68
## 9        Radiological    13
## 10 Sabotage Equipment   110
## 11            Unknown  9301
## 12            Vehicle    57
```

```
ggplot(aes(x = iyear, y = n), data = by_date_weap)+
  geom_line(aes(color = weaptype1_txt))
```

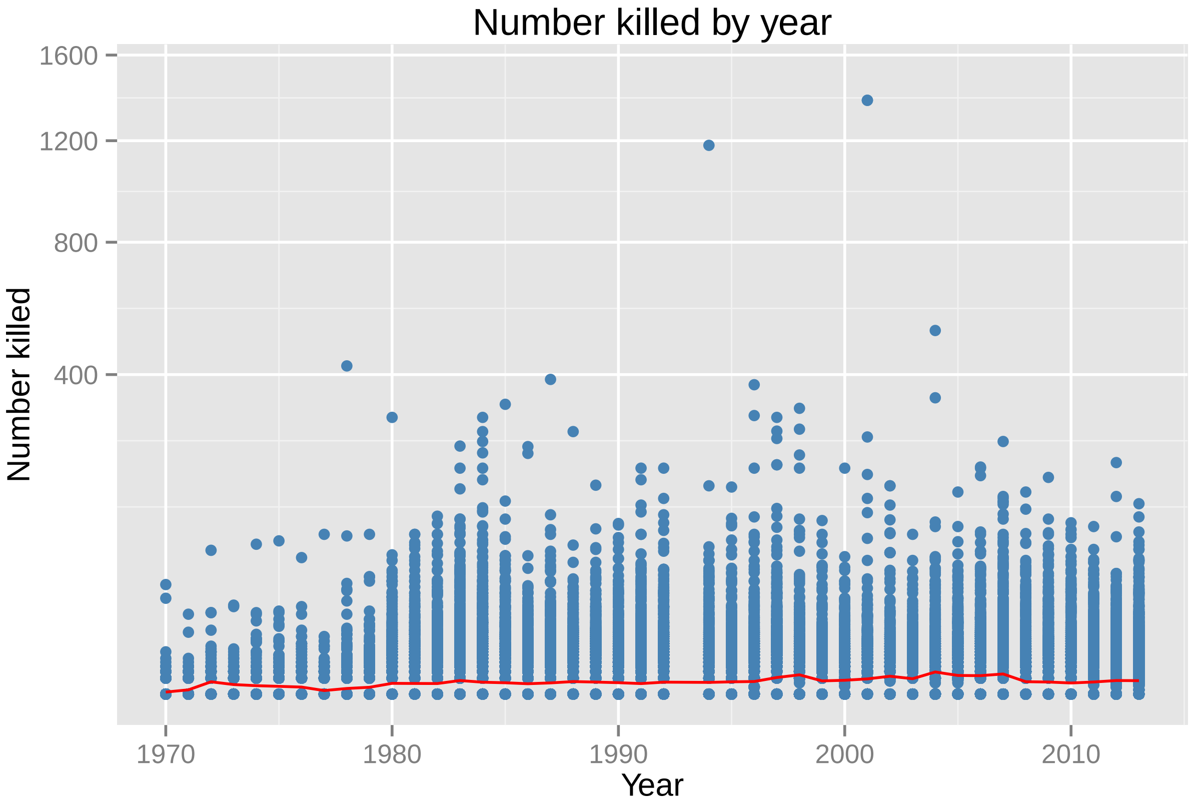


```
ggplot(aes(x = iyear, y = n), data = by_date_weap)+
  geom_line()+
  facet_wrap(~ weaptype1_txt)
```



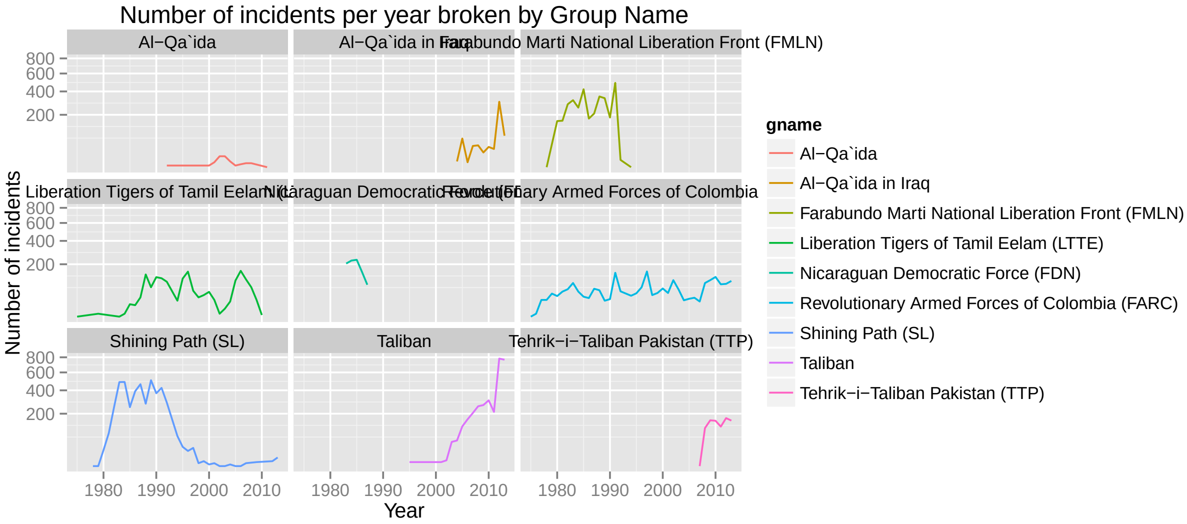
Final Visualizations and Summary

```
ggplot(aes(x = iyear, y = nkill), data = data)+
  geom_point(color = 'steelblue')+
  geom_line(stat = 'summary', fun.y = mean, color = 'red')+
  scale_y_sqrt(limits = c(0, 1500))+
  ggtitle("Number killed by year") + xlab("Year") + ylab('Number killed')
```



The mean in this plot (the red line) emphasises that while there are outliers with many hundreds killed in a single incident that the majority are much lower.

```
nkill_data <- head(nkill_data, n = 9)
data_by_year <- group_by(data[data$gname %in% nkill_data$gname, ], iyear, gname)
by_date_name <- summarise(data_by_year, n = n())

ggplot(aes(x = iyear, y = n), data = by_date_name)+
  geom_line(aes(color = gname))+
  facet_wrap(~gname)+
  scale_y_sqrt()+
  ggtitle("Number of incidents per year broken by Group Name")+
  xlab("Year") + ylab("Number of incidents")
```

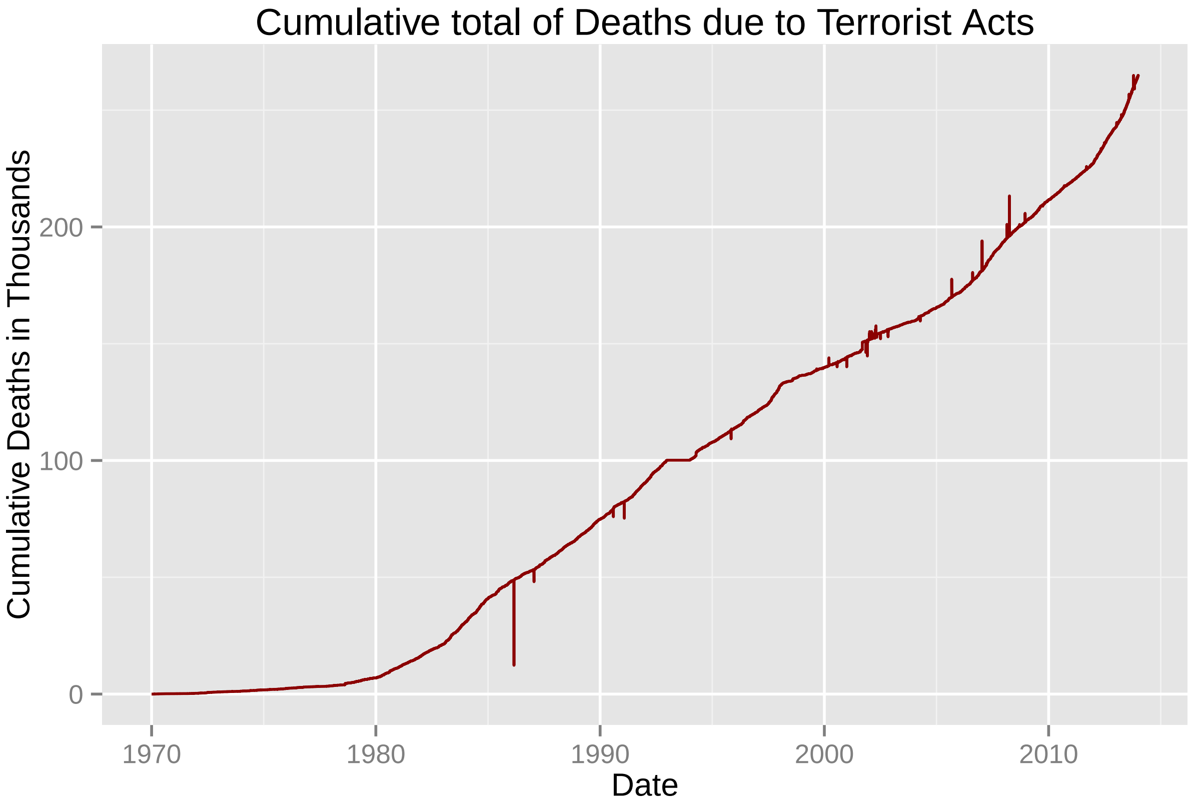


This plot shows the attack number by year of the 9 most casualty causing terrorist organizations. I notice that only three of the organizations have distributions across the entire time span. with most of them being relatively short lived but prolific.

I first decided to take a look at this data set, because I first started looking at the material for the course Exploratory Data Analysis with R in September hoping to prepare for the Data Analyst Nanodegree Program. I work in the security industry, and was living in the US on September 11th 2001. At work in the weeks leading up to the anniversary of that terrible day, there was a lot of talk about that incident and terrorism in general. I found the data set, and did some very basic plots to show people at work. I found the data set again on my computer when I set out to start this project, and thought I would use it for the project.

My last visualization is a pretty stark reminder of how terrible this pattern is.

```r
ggplot(aes(x = date, y = (cumsum(nkill)/1000)), data = data)+
  geom_line(color = "darkred")+
  ggtitle("Cumulative total of Deaths due to Terrorist Acts")+
  xlab("Date") + ylab("Cumulative Deaths in Thousands")
```



Reflections

I found this dataset a challenge to use. There are a many more variables than I was interested in taking a look at. The first hard decision was trying to figure out what I was going to look at in this project. Many of the variables also have too many possible values to graph properly. I feel too that the subset of interesting values I have taken a look at are too narrow, and don't really show anything interesting about the data. I feel that I have had many successes in the look at the dataset that I have taken. The first of which being figuring out how to construct the date variable from the available information. Secondly the structuring group by calls to put together an analsis of group involvement brought me a great deal of joy. As for future analysis, I plan to do a much more in depth analysis of this data using all of the tools that I now have available to me, including python's data analysis tools, and MongoDB. I would like to take a look at Incident types by region, possibly country, to see if there are any patterns there. I would like to analyse the groups, and see if I can group them and do an analysis of their spread, and efficacy over time, and many others.