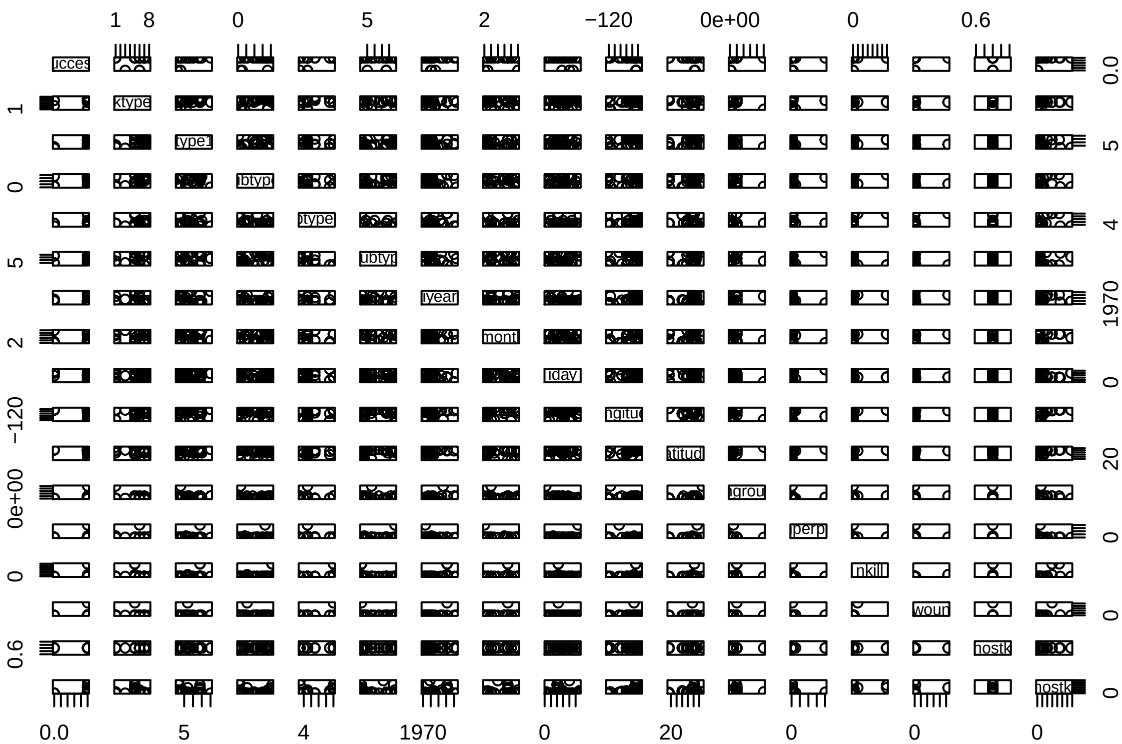
INFX 573 Exploratory Analyses

Timothy Pace, Boris Pavlov, Mike Stepanovic

2/21/2017

```r
# load packages
library(ggplot2)
library(dplyr)
library(maps)
library(mapproj)
library(fBasics)

# load data
gtd <- read.csv("globalterrorismdb_0616dist_US_ONLY.csv")
gtd_small <- select(gtd, success, attacktype1_txt, targtype1_txt,
                    targsubtype1_txt, weaptype1_txt, weapsubtype1_txt,
                    iyear, imonth, iday, longitude, latitude, ingroup,
                    nperps, nkill, nwound, ishostkid, nhostkid)

no.missing.data <- subset(gtd_small, nperps > 0 & nhostkid > 0 & ingroup > 0)
plot(no.missing.data)
```



1) How many unique observations to you have?

We have 2,693 unique observations:

```r
nrow(gtd_small)
```

```
## [1] 2693
```

2) What information/features/characteristics do you have for each observation?

We have a combination of data classes for continuous variables, as well as several categorical variables:

```r
str(gtd_small)
```

```
## 'data.frame':    2693 obs. of  17 variables:
##  $ success         : int  1 1 1 1 0 1 1 1 1 1 ...
##  $ attacktype1_txt : Factor w/ 9 levels "Armed Assault",..: 1 3 4 4 3 4 4 4 3 3 ...
##  $ targtype1_txt   : Factor w/ 22 levels "Abortion Related",..: 13 21 10 7 10 10 7 3 4 3 ...
##  $ targsubtype1_txt: Factor w/ 86 levels ".","Affiliated Institution",..: 56 15 41 22 38 41 22 72 73
##  $ weaptype1_txt   : Factor w/ 12 levels "Biological","Chemical",..: 5 3 6 6 3 6 6 6 3 3 ...
##  $ weapsubtype1_txt: Factor w/ 26 levels ".","Arson/Fire",..: 24 23 12 6 23 12 12 2 16 23 ...
##  $ iyear           : int  1970 1970 1970 1970 1970 1970 1970 1970 1970 1970 ...
##  $ imonth          : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ iday            : int  1 2 2 3 1 6 9 9 12 12 ...
##  $ longitude       : num  -89.2 -122.3 -89.4 -89.4 -89.7 ...
##  $ latitude        : num  37 37.8 43.1 43.1 43.5 ...
##  $ ingroup         : int  2373 -9 100003 100003 1231 453 453 3956 2373 612 ...
##  $ nperps          : int  -99 -99 1 1 NA -99 -99 -99 -99 -99 ...
##  $ nkill           : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ nwound          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ ishostkid       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ nhostkid        : int  NA NA NA NA NA NA NA NA NA NA ...
```

3) What are the min/max/mean/median/sd values for each of these features?

Continuous Variables

```r
# select only the continuous variables
gtd.continuous <- cbind(gtd$success, gtd$iyear, gtd$imonth, gtd$iday, gtd$longitude,
                        gtd$latitude, gtd$ingroup, gtd$nperps, gtd$nkill, gtd$nwound,
                        gtd$ishostkid, gtd$nhostkid)
# improve readability
colnames(gtd.continuous) <- c("success", "year", "month", "day", "longitude", "latitude",
                              "ingroup", "nperps", "nkill", "nwound", "ishostkid", "nhostkid")
# print descriptive stats
basicStats(gtd.continuous)[c("nobs", "NAs", "Minimum", "Maximum", "Mean", "Stdev"),]
```

```
##             success        year       month         day  longitude
## nobs    2693.000000 2693.00000 2693.000000 2693.000000 2693.00000
## NAs        0.000000    0.00000    0.000000    0.000000    1.00000
## Minimum    0.000000 1970.00000    1.000000    0.000000 -157.85833
## Maximum    1.000000 2015.00000   12.000000   31.000000  105.27055
## Mean       0.822131 1982.35351    6.209061   15.305607  -91.88302
## Stdev      0.382473   12.49159    3.400425    9.157802   21.98291
##            latitude    ingroup     nperps       nkill      nwound
## nobs    2693.000000   2693.000 2693.00000 2693.000000 2693.000000
## NAs        1.000000      0.000  982.00000   81.000000   95.000000
## Minimum   17.966072     -9.000  -99.00000    0.000000    0.000000
## Maximum   64.837778 100047.000  200.00000 1381.500000  751.000000
```

```
## Mean      36.650178   5012.613  -55.48451    1.369449    1.231332
## Stdev      7.416586   9706.309   50.46406   38.548154   20.471944
##           ishostkid    nhostkid
## nobs    2693.000000 2693.000000
## NAs      176.000000 2634.000000
## Minimum    0.000000  -99.000000
## Maximum    1.000000  135.000000
## Mean       0.023441   -5.016949
## Stdev      0.151328   49.642686
```

Categorical Variables

```r
# view top 10 levels for each categorical variable of interest
# need to reorder levels in attacktype1
gtd$attacktype1_txt <- factor(gtd_small$attacktype1_txt,
                              levels(gtd_small$attacktype1_txt)[c(3,4,1,2,8,6,7,5,9)])
summary(gtd["attacktype1_txt"], maxsum=10)
```

```
##                             attacktype1_txt
##  Bombing/Explosion                   :1369
##  Facility/Infrastructure Attack      : 802
##  Armed Assault                       : 234
##  Assassination                       : 126
##  Unarmed Assault                     :  58
##  Hostage Taking (Barricade Incident):  57
##  Hostage Taking (Kidnapping)         :  19
##  Hijacking                           :  17
##  Unknown                             :  11
```

```r
summary(gtd["targsubtype1_txt"], maxsum=10)
```

```
##                                targsubtype1_txt
##  Clinics                                : 238
##  Bank/Commerce                          : 220
##  Government Building/Facility/Office    : 197
##  Retail/Grocery/Bakery                  : 188
##  School/University/Educational Building: 129
##  .                                      : 120
##  Military Recruiting Station/Academy    :  83
##  Place of Worship                       :  81
##  Industrial/Textiles/Factory            :  75
##  (Other)                                :1362
```


```r
summary(gtd["targsubtype1_txt"], maxsum=10)
```

```
##                                targsubtype1_txt
##  Clinics                                : 238
##  Bank/Commerce                          : 220
##  Government Building/Facility/Office    : 197
##  Retail/Grocery/Bakery                  : 188
##  School/University/Educational Building: 129
##  .                                      : 120
##  Military Recruiting Station/Academy    :  83
##  Place of Worship                       :  81
##  Industrial/Textiles/Factory            :  75
```

```
##  (Other)                               :1362
```

```r
summary(gtd["weaptype1_txt"], maxsum=10)
```

```
##                   weaptype1_txt
##  Explosives/Bombs/Dynamite:1377
##  Incendiary               : 794
##  Firearms                 : 359
##  Unknown                  :  50
##  Melee                    :  30
##  Biological               :  24
##  Sabotage Equipment       :  18
##  Other                    :  16
##  Chemical                 :  10
##  (Other)                  :  15
```

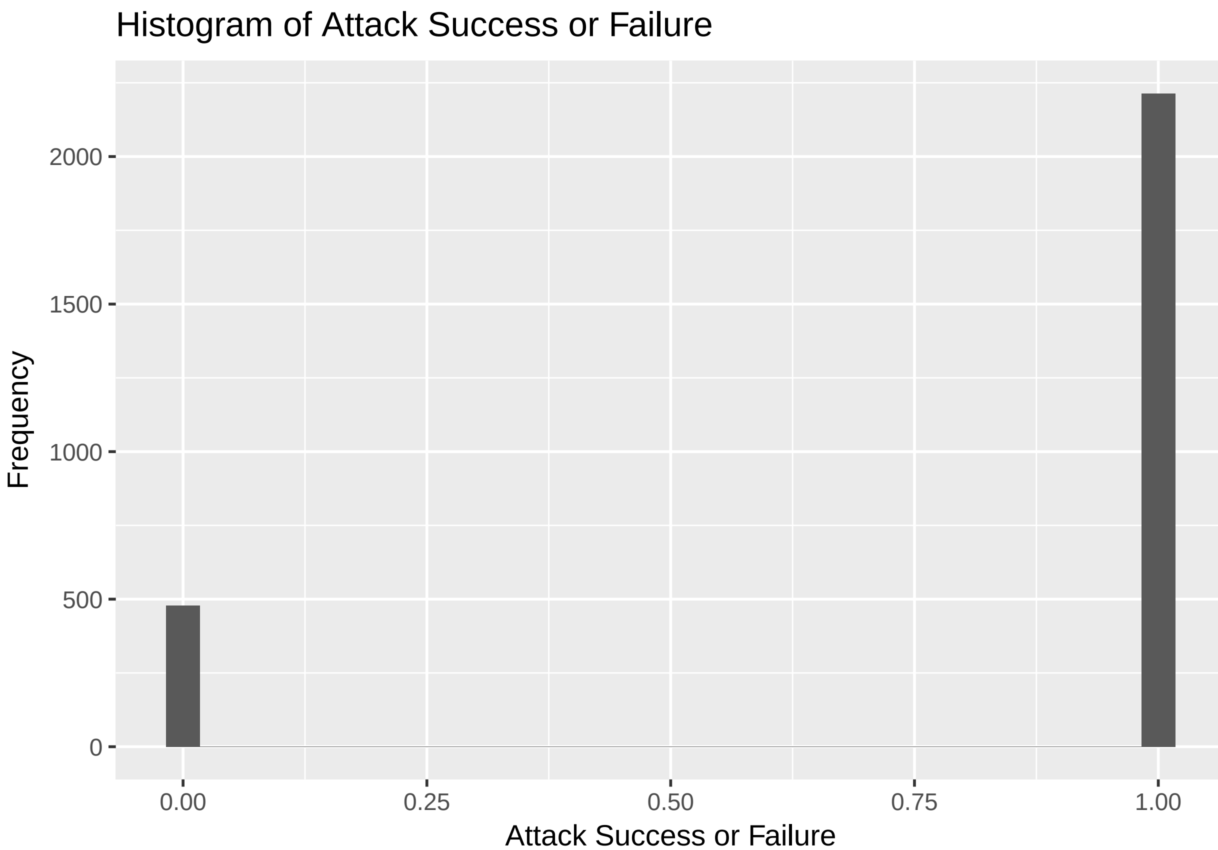
```r
summary(gtd["weapsubtype1_txt"], maxsum=10)
```

```
##                     weapsubtype1_txt
##  Unknown Explosive Type      :842
##  .                           :264
##  Arson/Fire                  :256
##  Molotov Cocktail/Petrol Bomb:211
##  Gasoline or Alcohol         :191
##  Other Explosive Type        :152
##  Handgun                     :143
##  Dynamite/TNT                :132
##  Time Fuse                   :129
##  (Other)                     :373
```

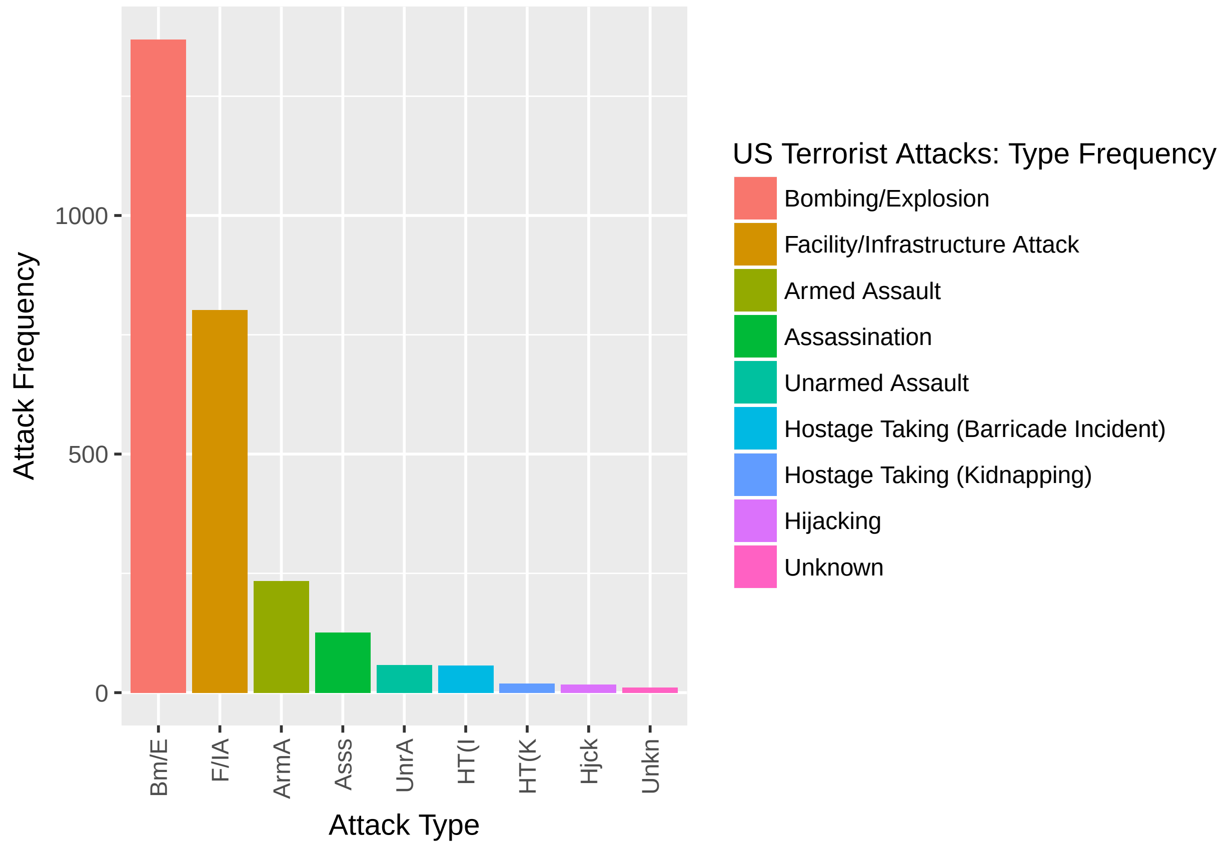
4) What is the distribution of the core features (show a histogram)?

```r
# success
ggplot(gtd, aes(x = success)) + geom_histogram() +
  labs(x = "Attack Success or Failure", y = "Frequency",
       title = "Histogram of Attack Success or Failure")
```

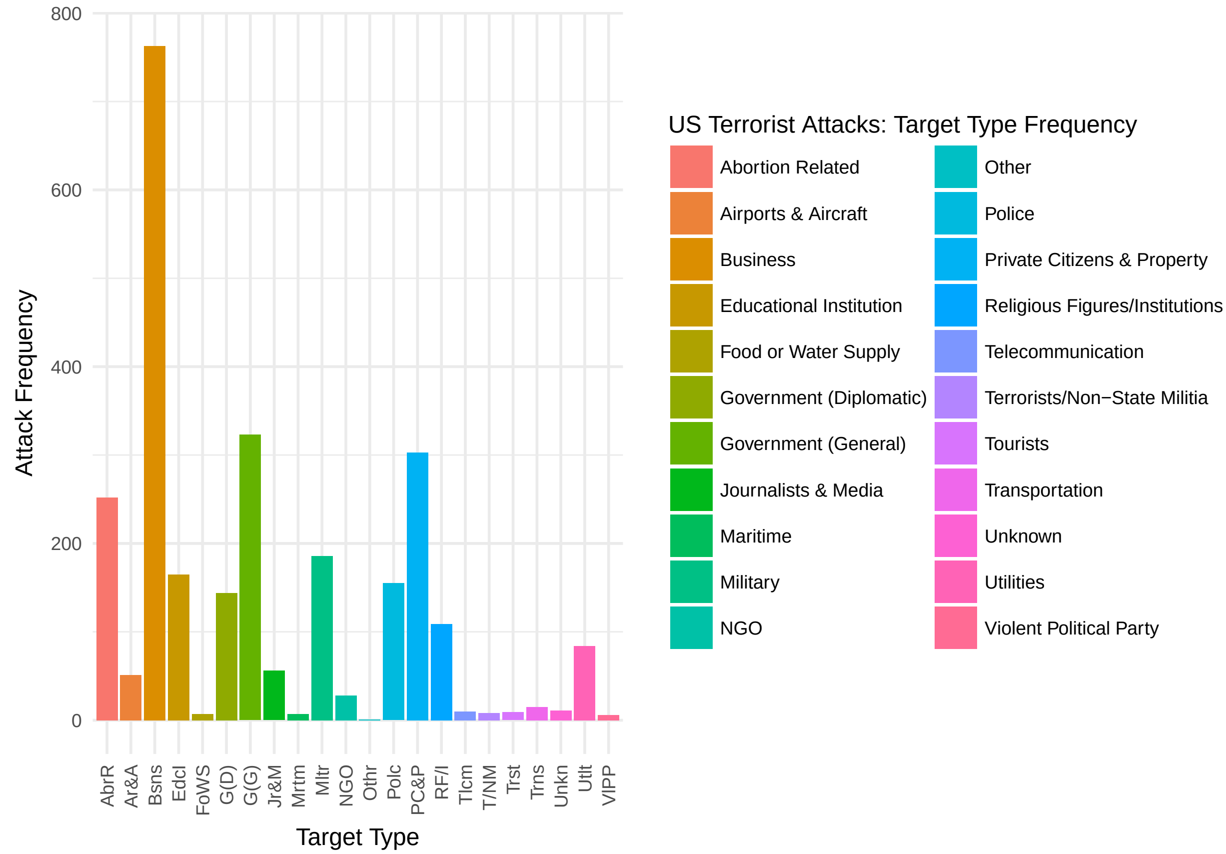
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



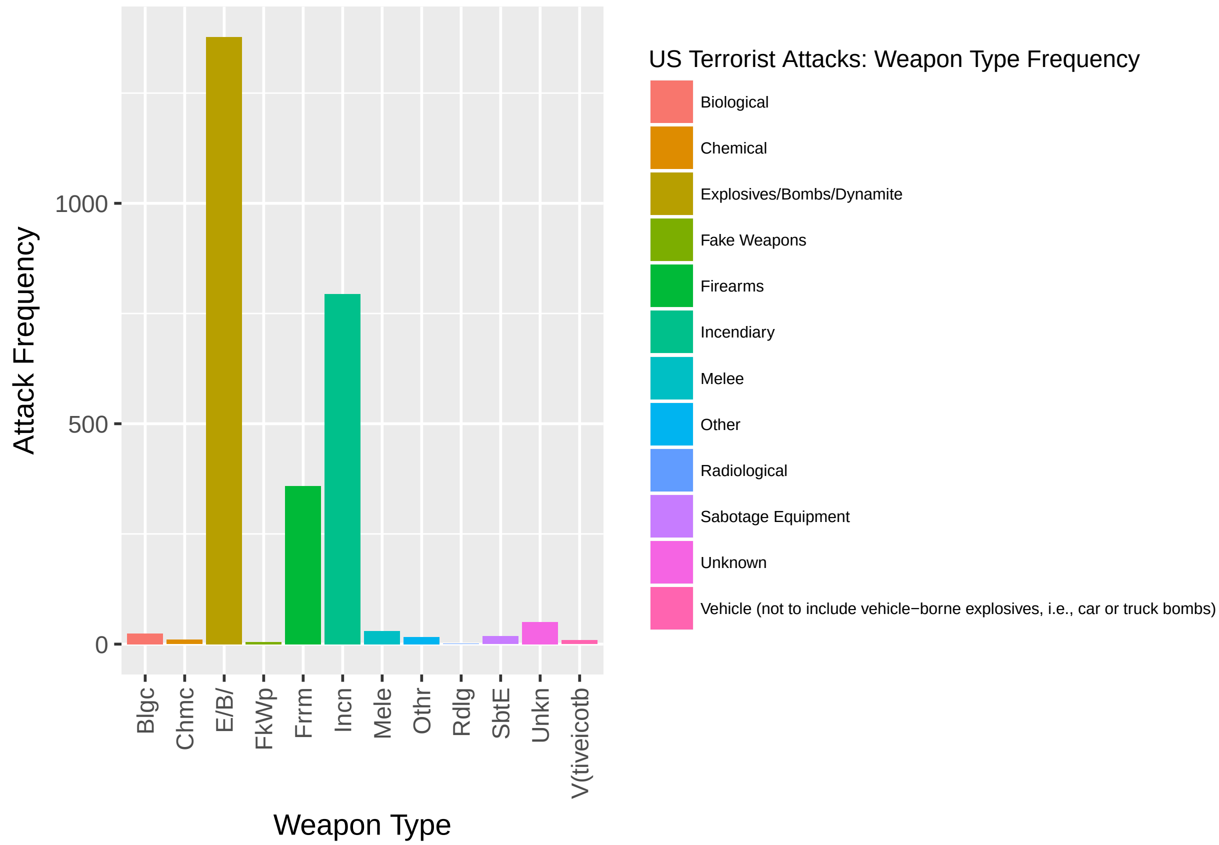

```
# attack type
ggplot(gtd, aes(x = attacktype1_txt, fill = attacktype1_txt)) + geom_bar() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  scale_x_discrete(labels = abbreviate) +
  labs(x = "Attack Type", y = "Attack Frequency") +
  guides(fill=guide_legend(title="US Terrorist Attacks: Type Frequency"))
```



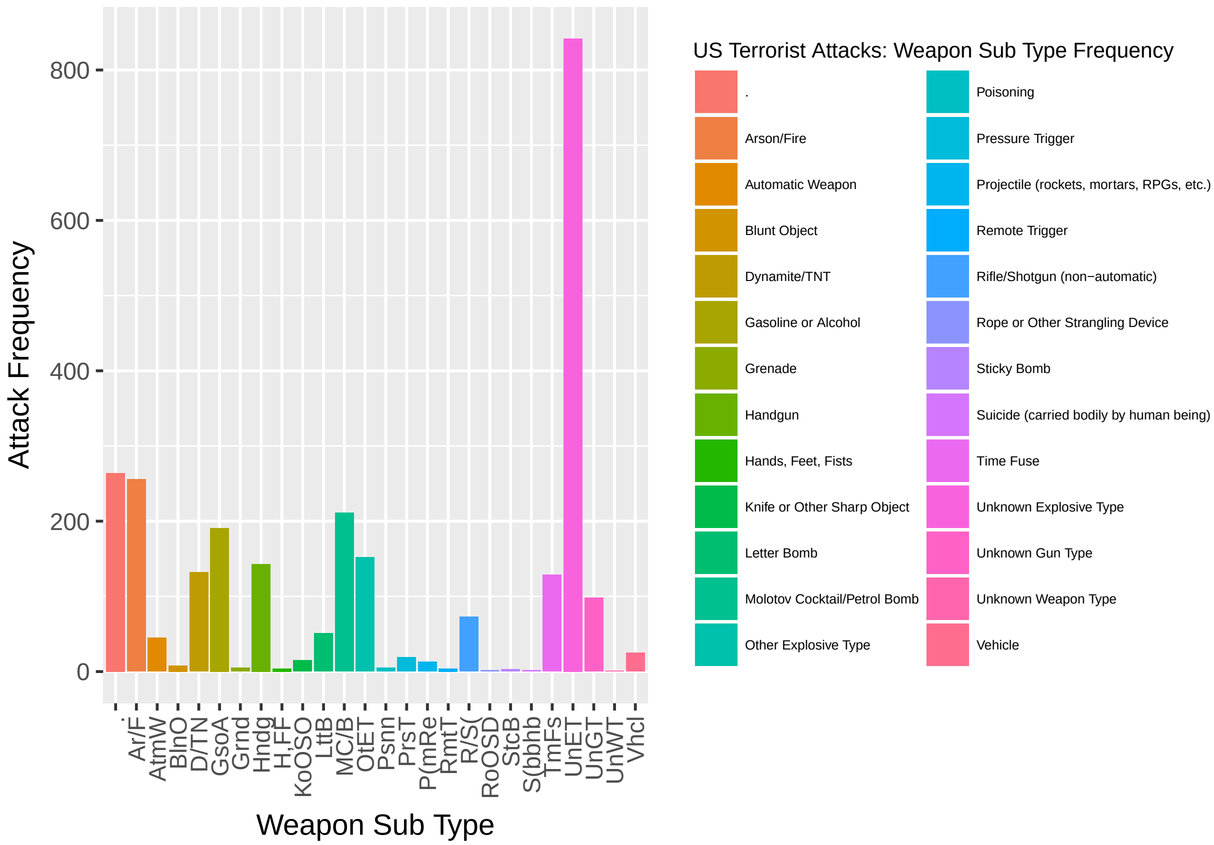

```r
# target type
ggplot(gtd, aes(x = targtype1_txt)) + geom_bar(aes(fill = targtype1_txt)) +
  theme_minimal(base_size = 9) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5),
        legend.text = element_text(size = 7)) +
  scale_x_discrete(labels = abbreviate) + labs(x = "Target Type", y = "Attack Frequency") +
  guides(fill = guide_legend(title = "US Terrorist Attacks: Target Type Frequency"))
```




```r
# weapon type
ggplot(gtd, aes(x = weaptype1_txt, fill = weaptype1_txt)) + geom_bar() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5),
  legend.text = element_text(size = 6), legend.title=element_text(size=9)) +
  scale_x_discrete(labels = abbreviate) + labs(x = "Weapon Type", y = "Attack Frequency") +
  guides(fill = guide_legend(title = "US Terrorist Attacks: Weapon Type Frequency"))
```

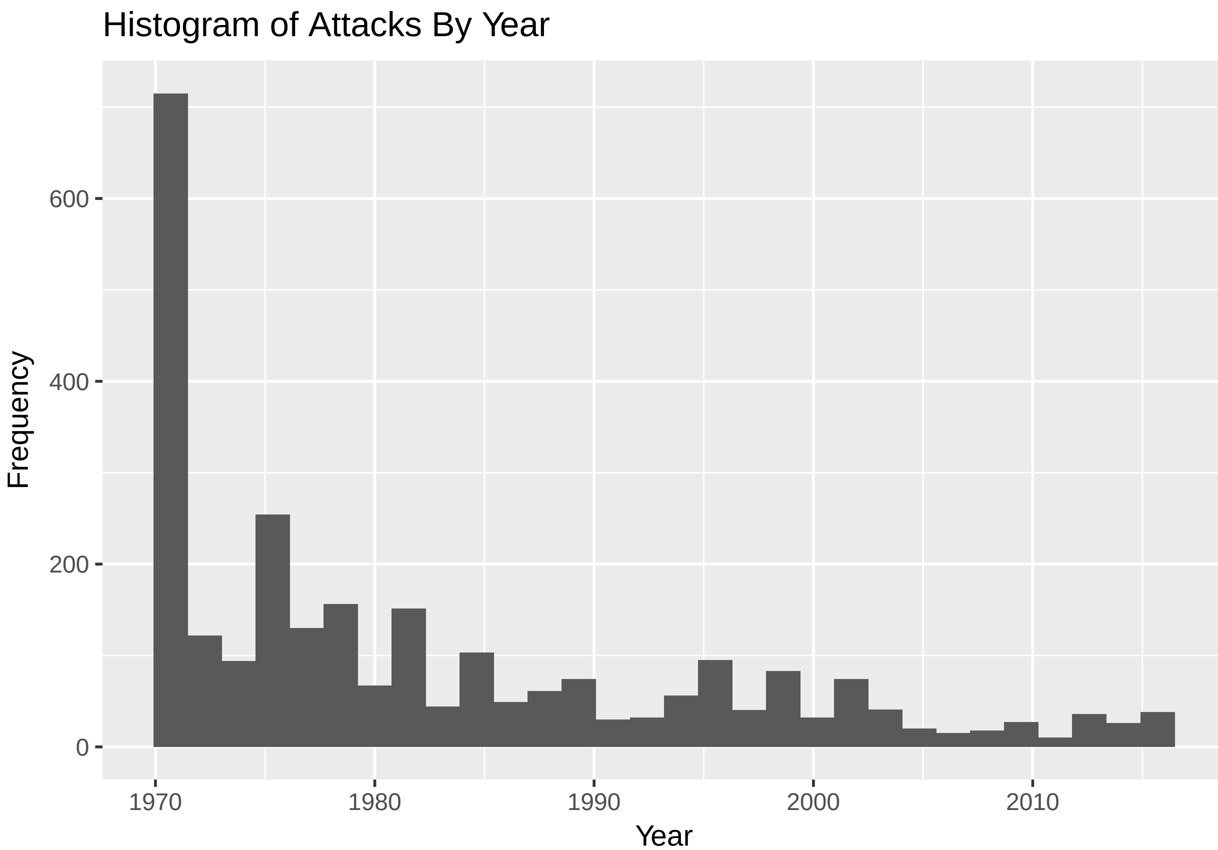



```r
# weapon subtype
ggplot(gtd, aes(x = weapsubtype1_txt, fill = weapsubtype1_txt)) + geom_bar() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5),
  legend.text = element_text(size = 5), legend.title = element_text(size = 8)) +
  scale_x_discrete(labels = abbreviate) +
  guides(fill = guide_legend(title = "US Terrorist Attacks: Weapon Sub Type Frequency")) +
  labs(x = "Weapon Sub Type", y = "Attack Frequency")
```



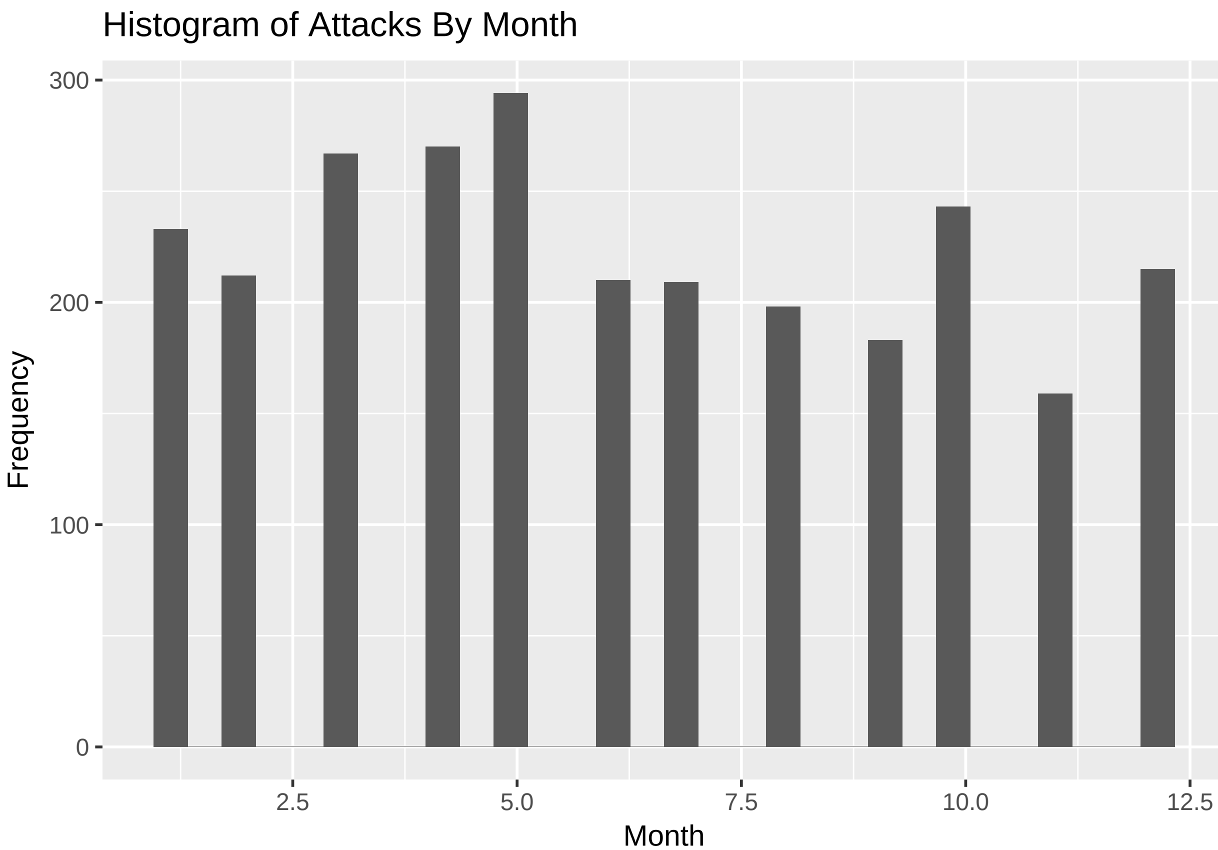

```
# year
ggplot(gtd, aes(x = iyear)) + geom_histogram() +
  labs(title = "Histogram of Attacks By Year", x = "Year", y = "Frequency")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



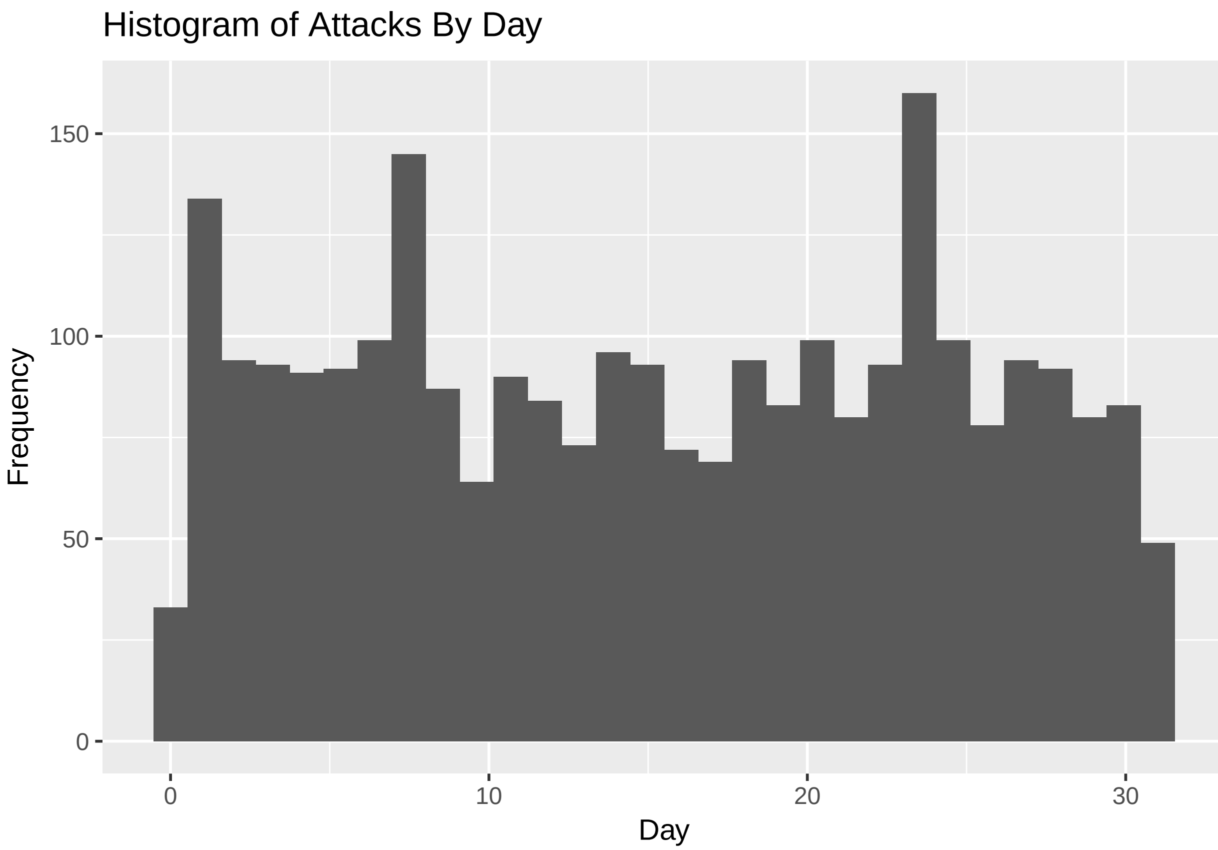

```
# month
  ggplot(gtd, aes(x = imonth)) + geom_histogram() +
  labs(title = "Histogram of Attacks By Month", x = "Month", y = "Frequency")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



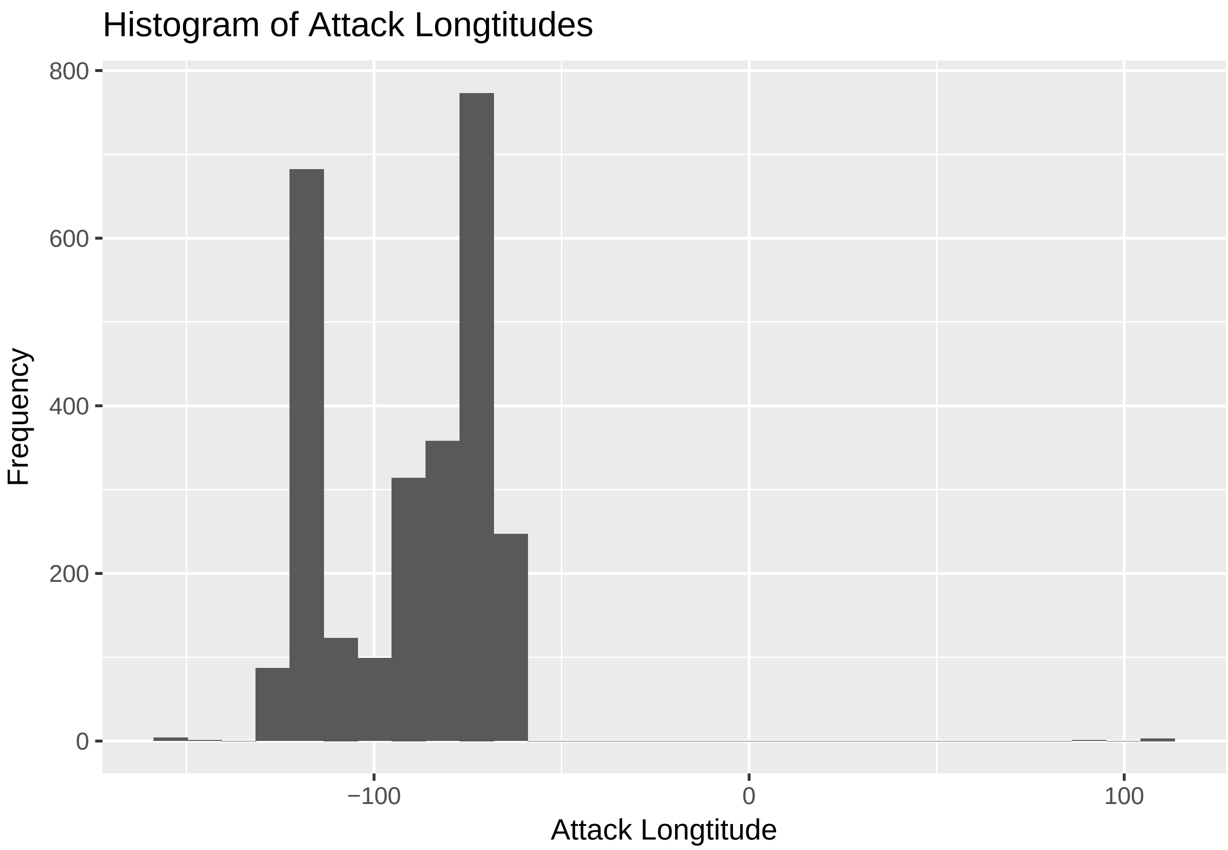

```r
# day
ggplot(gtd, aes(x = iday)) + geom_histogram() +
  labs(title = "Histogram of Attacks By Day", x = "Day", y = "Frequency")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```r
# longitude
ggplot(gtd, aes(x = longitude)) + geom_histogram() +
  labs(title = "Histogram of Attack Longtitudes", x = "Attack Longtitude", y = "Frequency")
```

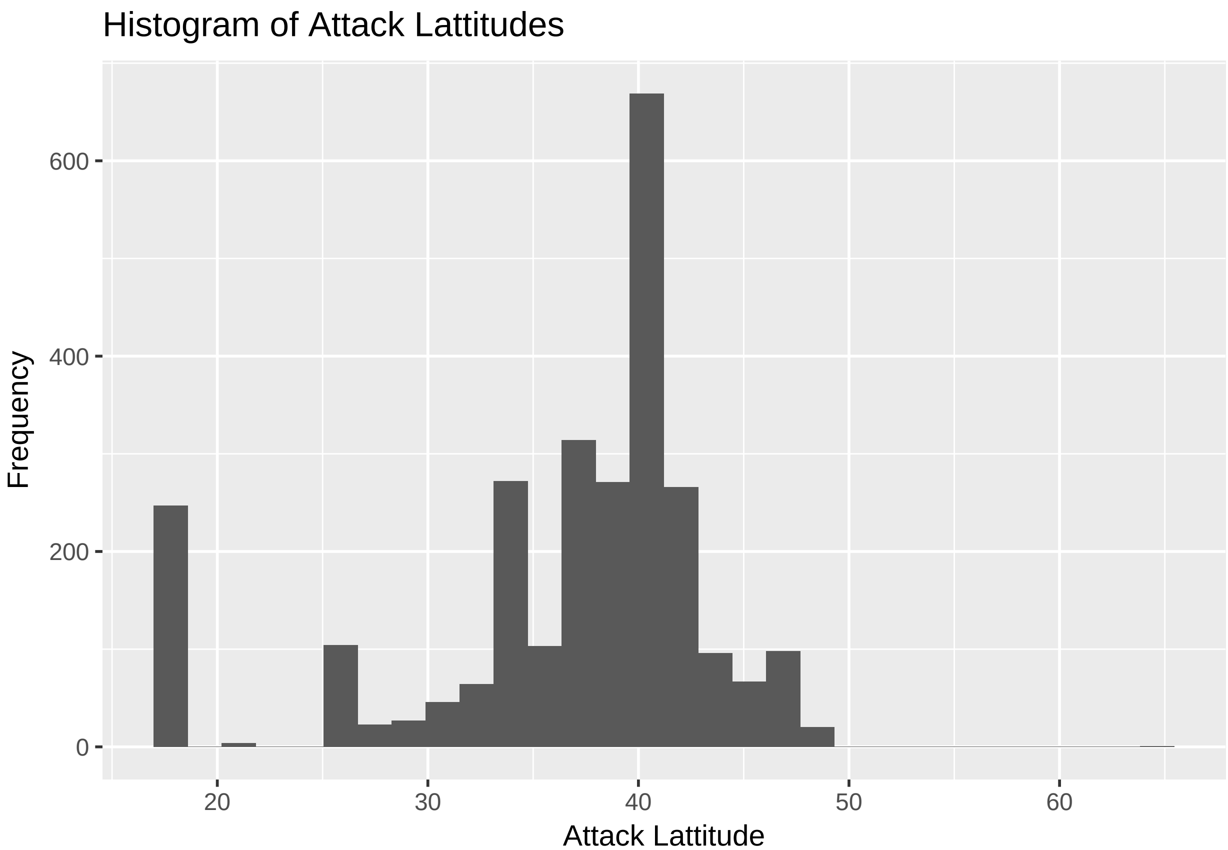
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 1 rows containing non-finite values (stat_bin).
```




```r
# latitude
ggplot(gtd, aes(x = latitude)) + geom_histogram() +
  labs(title = "Histogram of Attack Lattitudes", x = "Attack Lattitude",
  y = "Frequency")
```

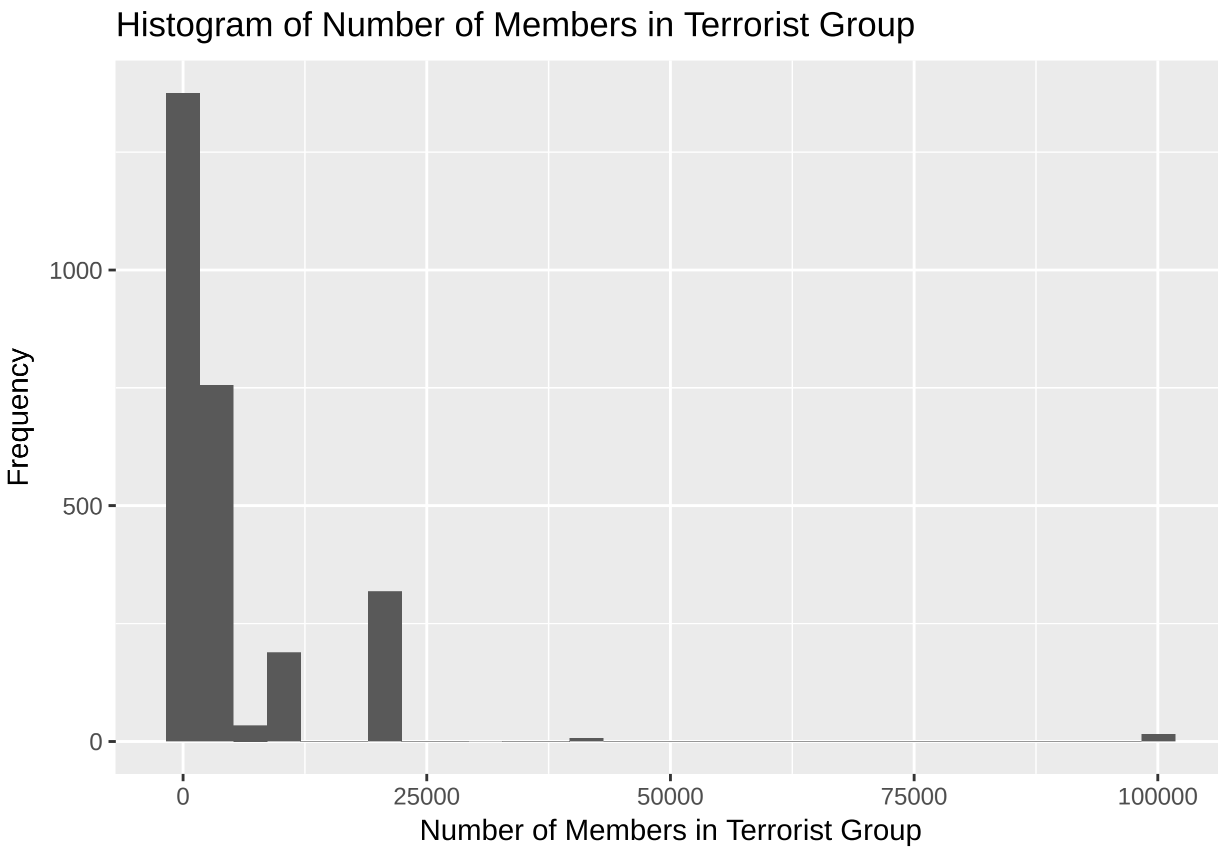
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 1 rows containing non-finite values (stat_bin).
```



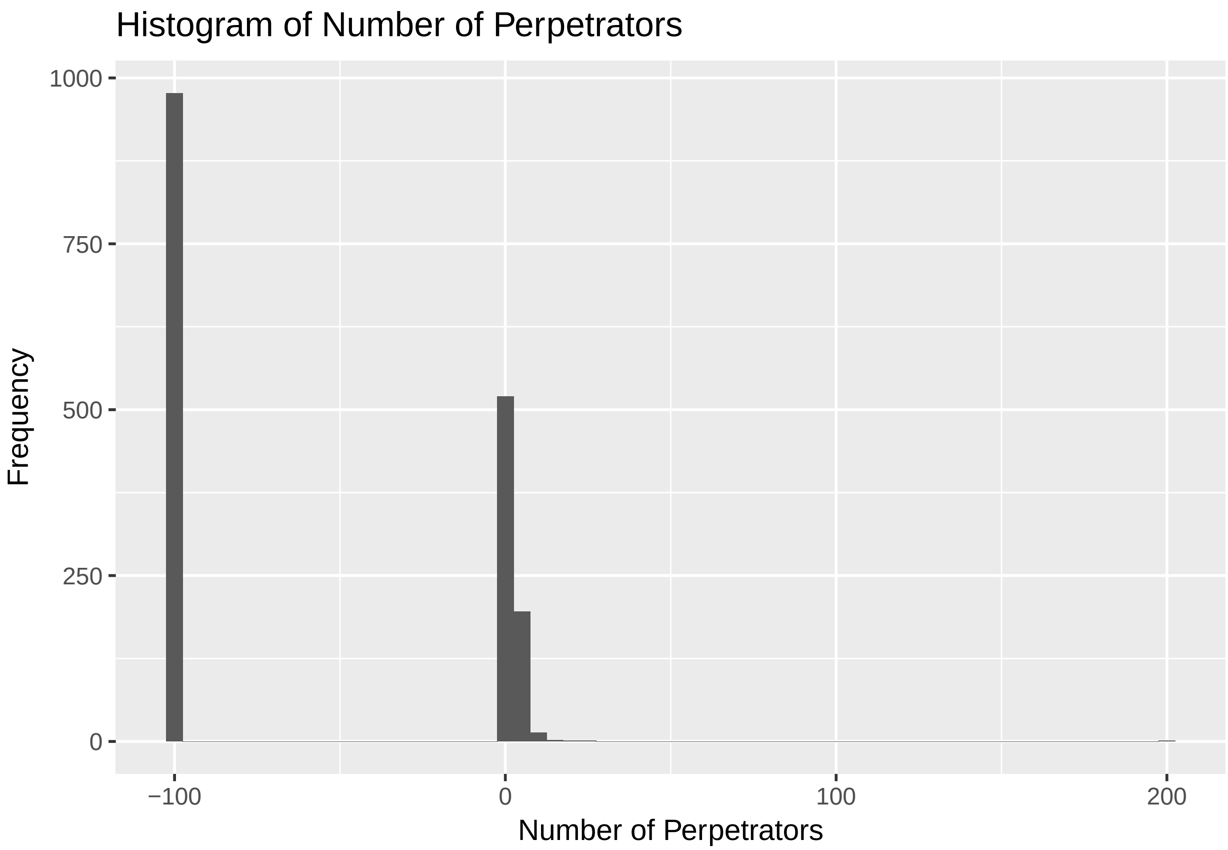

```r
# ingroup
ggplot(gtd, aes(x = ingroup)) + geom_histogram() +
  labs(title = "Histogram of Number of Members in Terrorist Group",
  x = "Number of Members in Terrorist Group", y = "Frequency")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```r
# nperp
ggplot(gtd, aes(x = nperps)) + geom_histogram(binwidth = 5) +
  labs(title = "Histogram of Number of Perpetrators",
  x = "Number of Perpetrators", y = "Frequency")
```

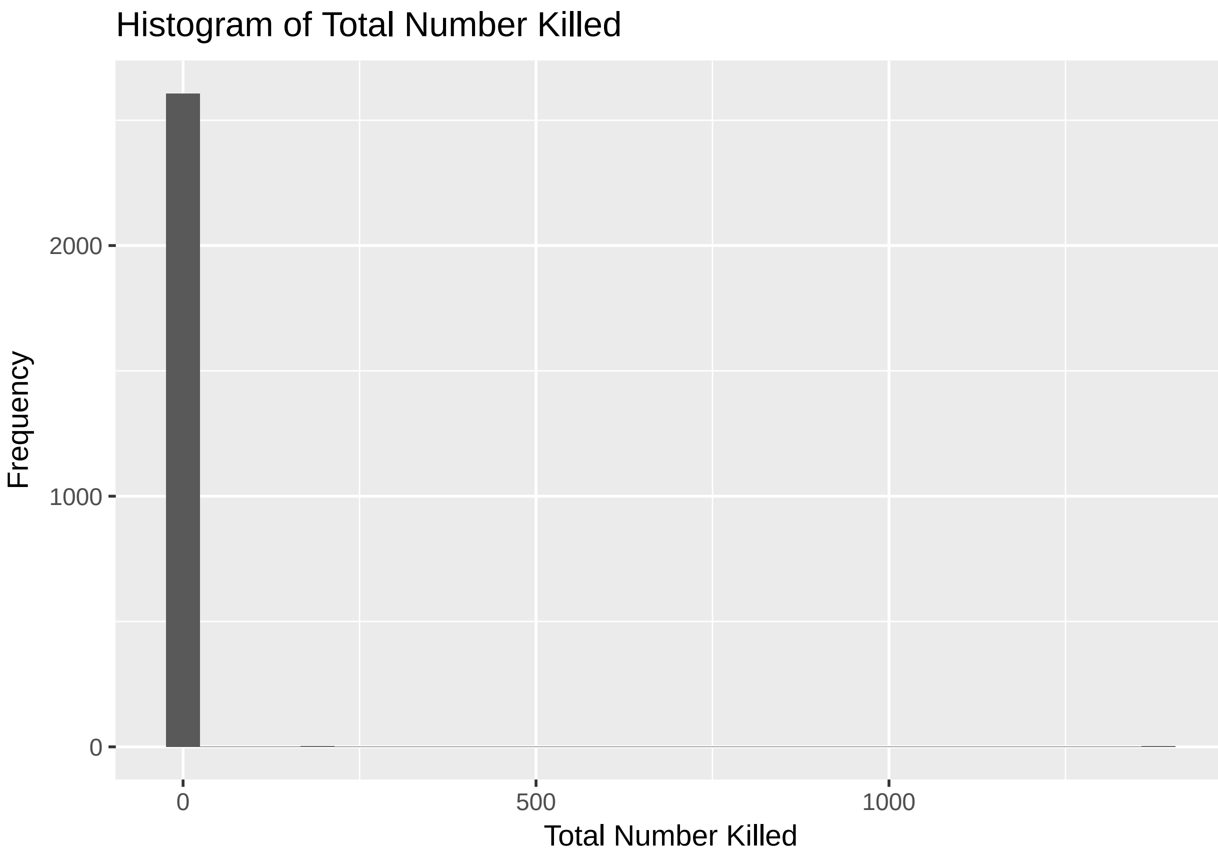
```
## Warning: Removed 982 rows containing non-finite values (stat_bin).
```




```r
# nkill
ggplot(gtd, aes(x = nkill)) + geom_histogram() +
  labs(title = "Histogram of Total Number Killed",
  x = "Total Number Killed", y = "Frequency")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

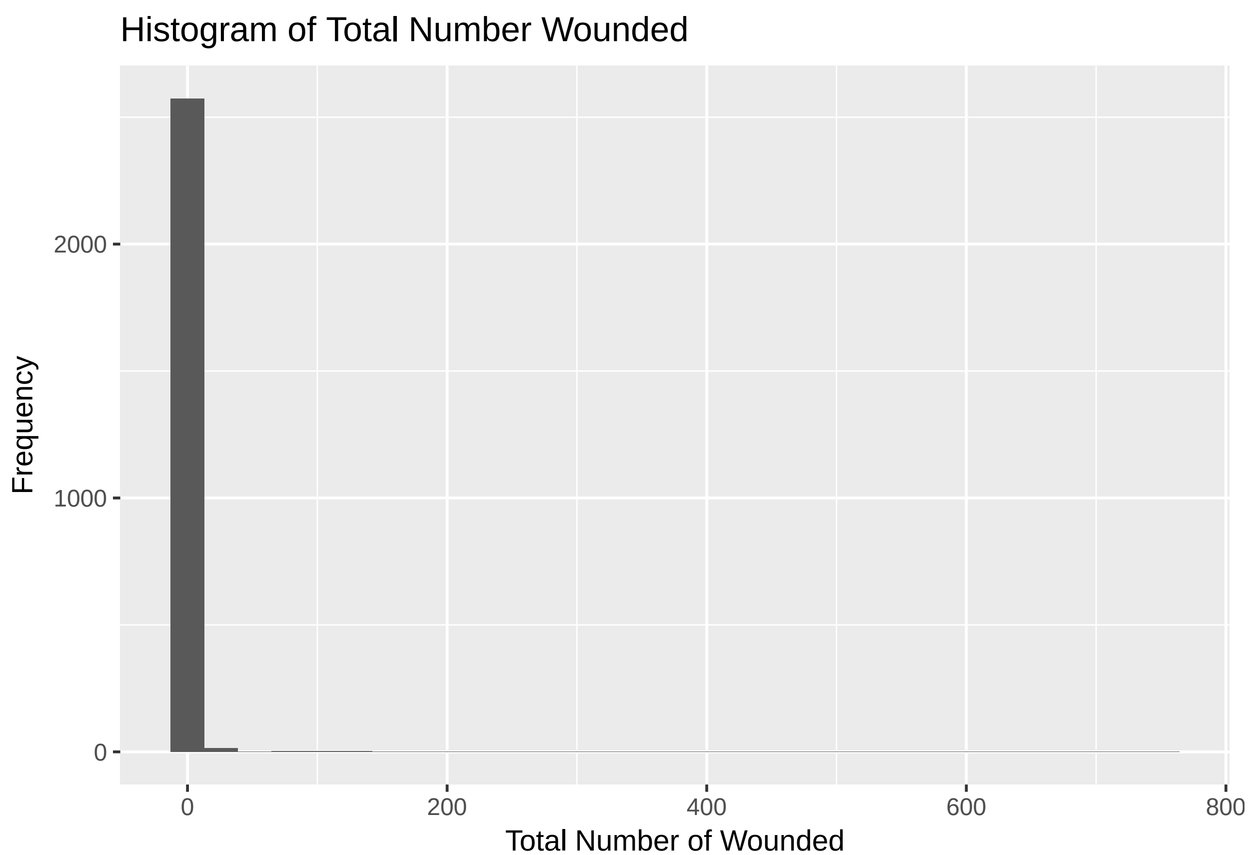
```
## Warning: Removed 81 rows containing non-finite values (stat_bin).
```




```r
# nwound
ggplot(gtd, aes(x = nwound)) + geom_histogram() +
  labs(title = "Histogram of Total Number Wounded",
  x = "Total Number of Wounded", y = "Frequency")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

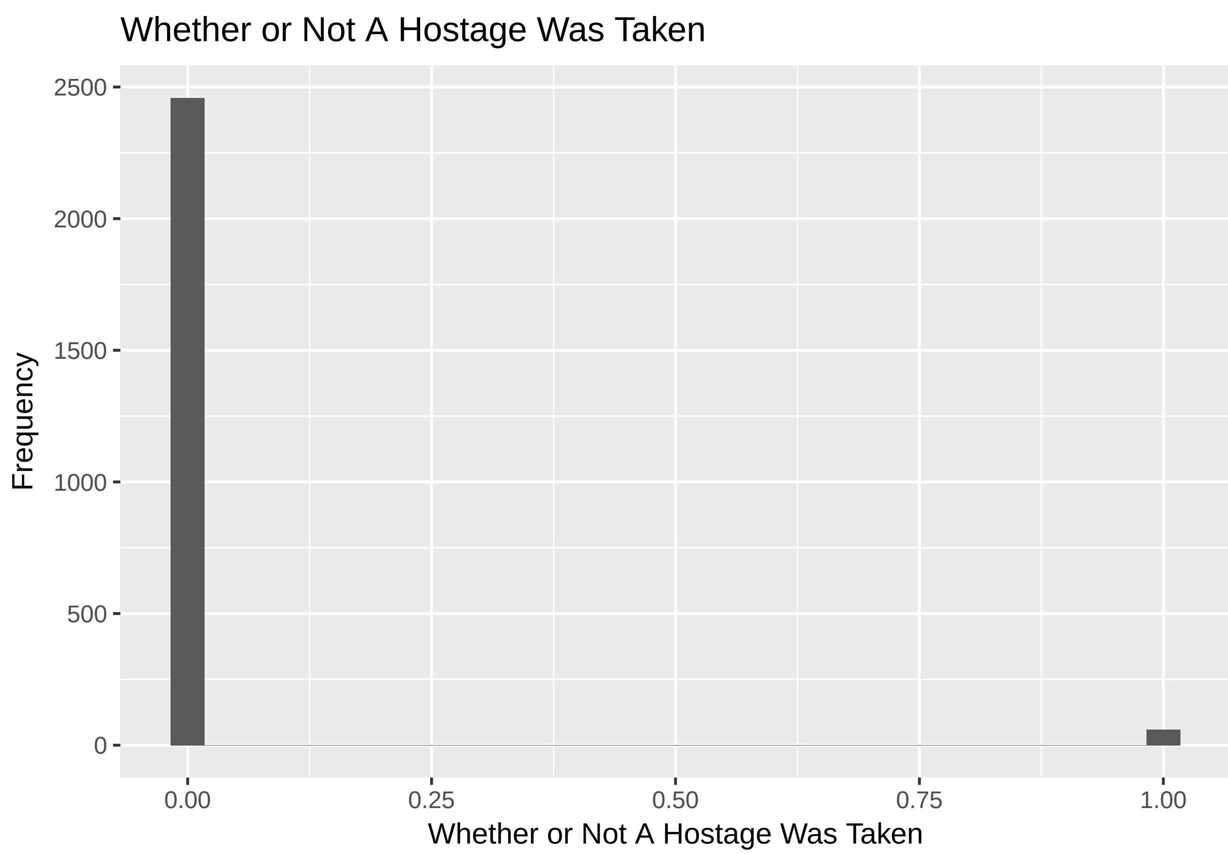
```
## Warning: Removed 95 rows containing non-finite values (stat_bin).
```




```r
# ishostkid
ggplot(gtd, aes(x = ishostkid)) + geom_histogram() +
  labs(x = "Whether or Not A Hostage Was Taken", y = "Frequency",
       title = "Whether or Not A Hostage Was Taken")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

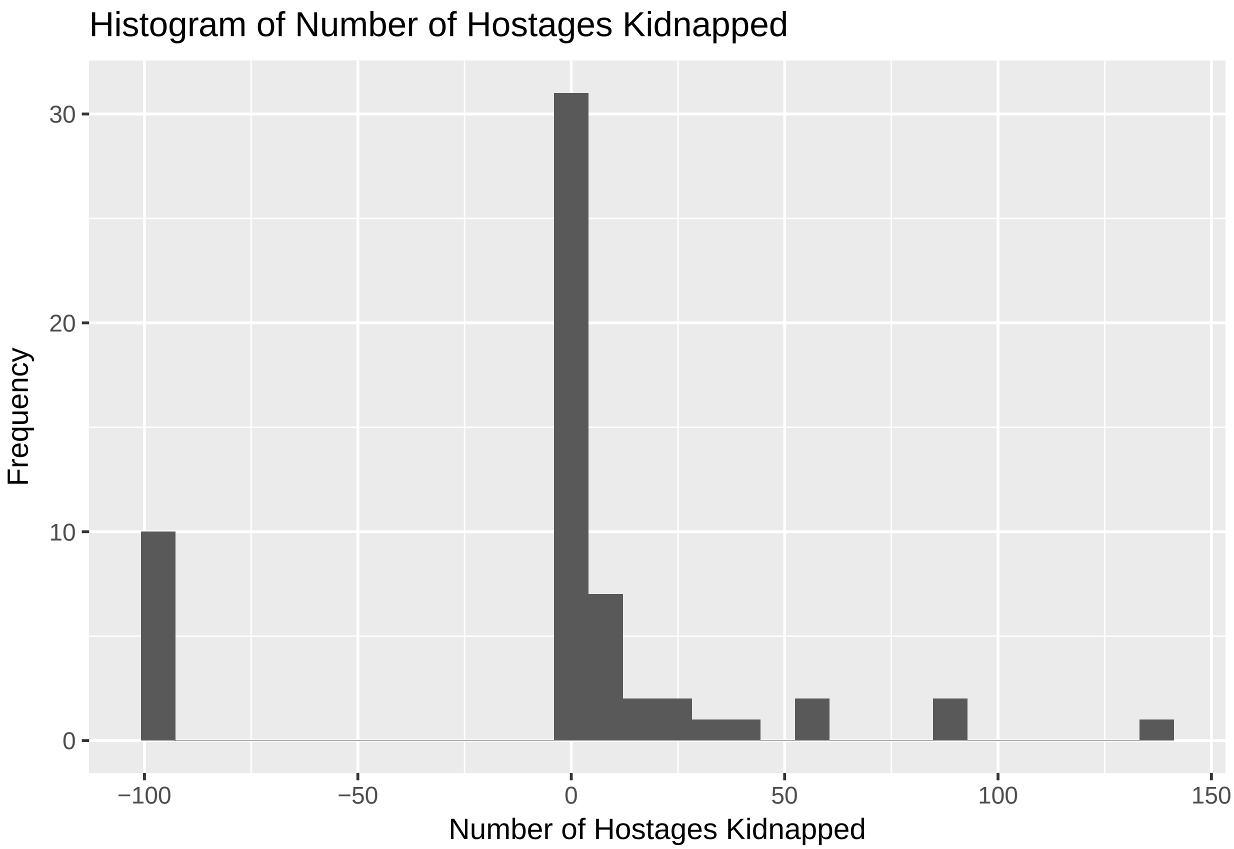
```
## Warning: Removed 176 rows containing non-finite values (stat_bin).
```




```r
# nhostkid
ggplot(gtd, aes(x = nhostkid)) + geom_histogram() +
  labs(title = "Histogram of Number of Hostages Kidnapped",
  x = "Number of Hostages Kidnapped", y = "Frequency")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 2634 rows containing non-finite values (stat_bin).
```



5) Are there obvious trends in the data (over time, across subgroups, etc.), and are the differences statistically significant?

```
# subset data of interest
gtd.noWTC <- subset(gtd_small, nkill < 1000)

gtd.cont2.df <- select(no.missing.data, success, ingroup, nperps, nkill, nwound, nhostkid)
gtd.cont2.df.nowtc <- subset(gtd.cont2.df, nkill < 1000)
cor(gtd.cont2.df.nowtc)
```

```
##              success     ingroup      nperps       nkill      nwound
## success  1.00000000  0.13435731  0.03963195  0.05856975  0.04874542
## ingroup  0.13435731  1.00000000 -0.02122551  0.12804039  0.10177169
## nperps   0.03963195 -0.02122551  1.00000000 -0.01883956 -0.00500500
## nkill    0.05856975  0.12804039 -0.01883956  1.00000000  0.97193834
## nwound   0.04874542  0.10177169 -0.00500500  0.97193834  1.00000000
## nhostkid 0.13711434  0.05140481 -0.03419850  0.30689532  0.28303077
##             nhostkid
## success   0.13711434
## ingroup   0.05140481
## nperps   -0.03419850
## nkill     0.30689532
## nwound    0.28303077
## nhostkid  1.00000000
```

```r
# perform chi squared test between success and nkilled
test1 <- table(gtd.cont2.df.nowtc$nkill, gtd.cont2.df.nowtc$success)
chisq.test(test1)
```

```
## Warning in chisq.test(test1): Chi-squared approximation may be incorrect

##
##  Pearson's Chi-squared test
##
## data:  test1
## X-squared = 0.975, df = 4, p-value = 0.9136
```

```r
# perform chi squared test between attack type and success
test2 <- table(gtd.noWTC$attacktype1_txt, gtd.noWTC$success)
chisq.test(test2)
```

```
## Warning in chisq.test(test2): Chi-squared approximation may be incorrect

##
##  Pearson's Chi-squared test
##
## data:  test2
## X-squared = 135.38, df = 8, p-value < 2.2e-16
```

```r
test3 <- table(gtd.noWTC$attacktype1_txt, gtd.noWTC$ishostkid)
chisq.test(test3)
```

```
## Warning in chisq.test(test3): Chi-squared approximation may be incorrect

##
##  Pearson's Chi-squared test
##
## data:  test3
## X-squared = 1387.4, df = 8, p-value < 2.2e-16
```

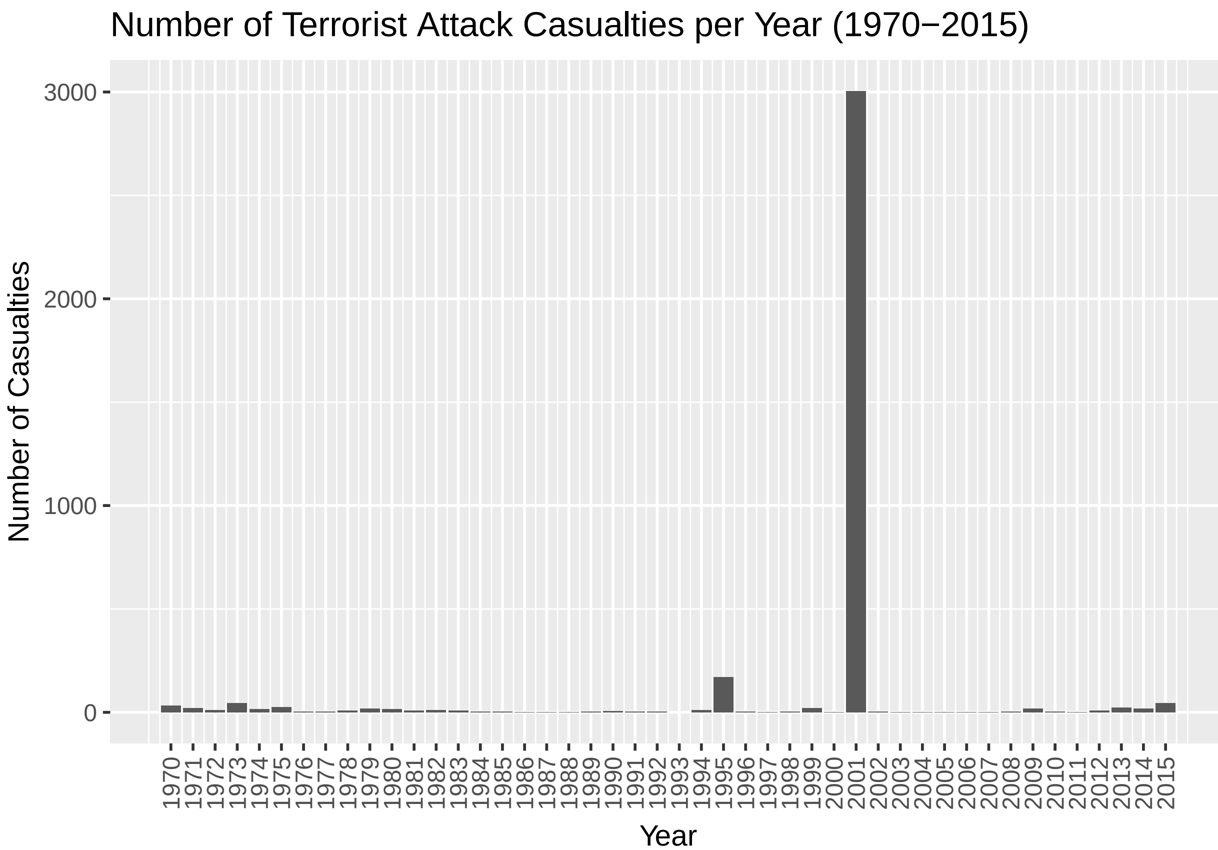
There is a high correlation between the number of casualties in an attack (`nkill`) and number wounded (`nwounded`) and we would not include both variables in a regression model.

We performed chi square tests for a statistically significant relationship between number of casualties and success of attack (not significant), type of attack and success of attack (significant), and the type of attack and whether hostages were taken (significant).

We've selected a number of interesting trends based on our primary and secondary hypotheses. First, we can examine how casualties have changed over time:

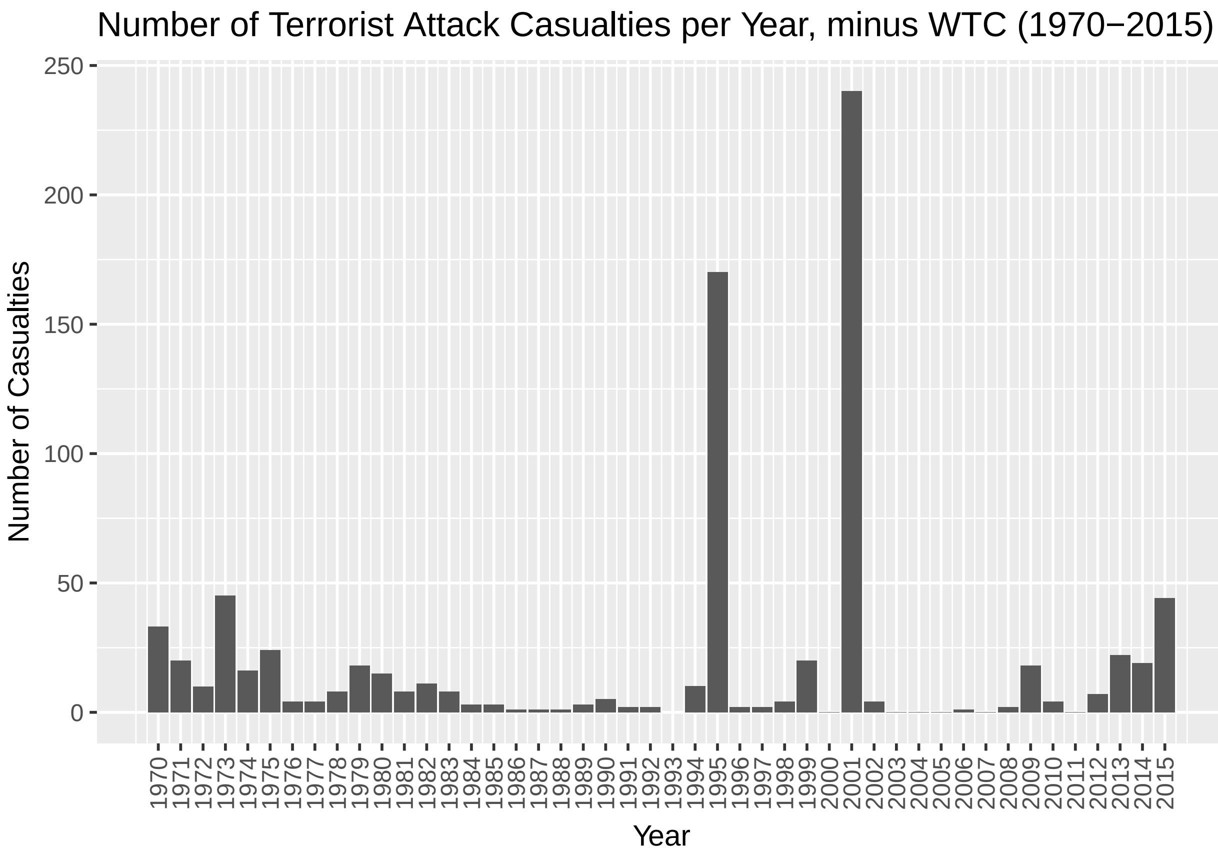
```r
# plot casualties per year
gg.months <- ggplot(gtd_small, aes(iyear, nkill))
gg.months + geom_bar(stat = "identity") + xlab("Year") + ylab("Number of Casualties") +
  ggtitle("Number of Terrorist Attack Casualties per Year (1970-2015)") + theme_gray() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  scale_x_continuous(breaks = 1970:2015)
```

```
## Warning: Removed 81 rows containing missing values (position_stack).
```



It appears that the 9/11 WTC attacks are a significant outlier. Here's the same chart, minus the WTC attacks:

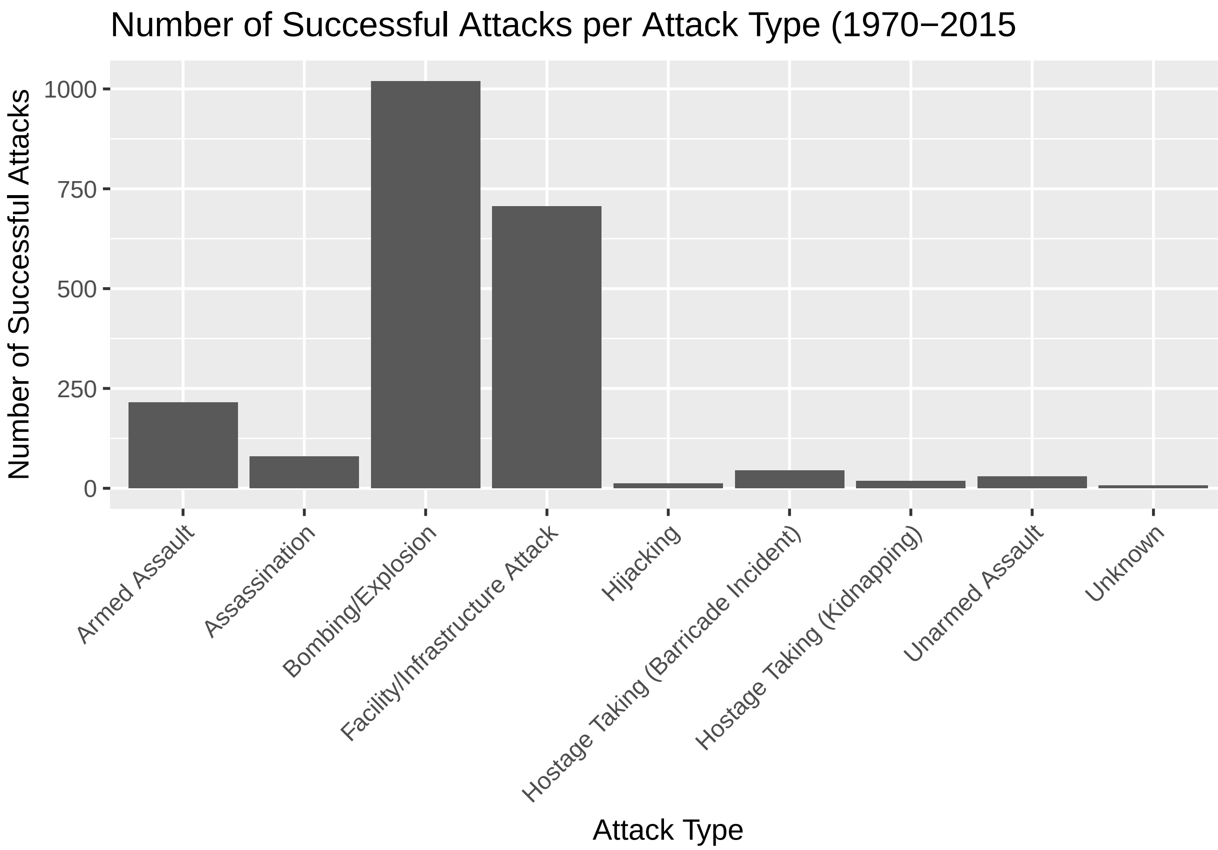
```
# plot
gg.months <- ggplot(gtd.noWTC, aes(iyear, nkill))
gg.months + geom_bar(stat="identity") + xlab("Year") + ylab("Number of Casualties") +
  ggtitle("Number of Terrorist Attack Casualties per Year, minus WTC (1970-2015)") +
  scale_x_continuous(breaks=1970:2015) + theme_gray() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))
```



This seems much more helpful for thinking about our model.

Here's the number of successful attacks per attack type (again removing the WTC):

```
# plot attack success by attack type, excluding WTC
gg.success.att <- ggplot(gtd.noWTC, aes(attacktype1_txt, success))
gg.success.att + geom_bar(stat="identity") + ylab("Number of Successful Attacks") +
  ggtitle("Number of Successful Attacks per Attack Type (1970-2015") + xlab("Attack Type") +
  theme_gray() + theme(axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1))
```

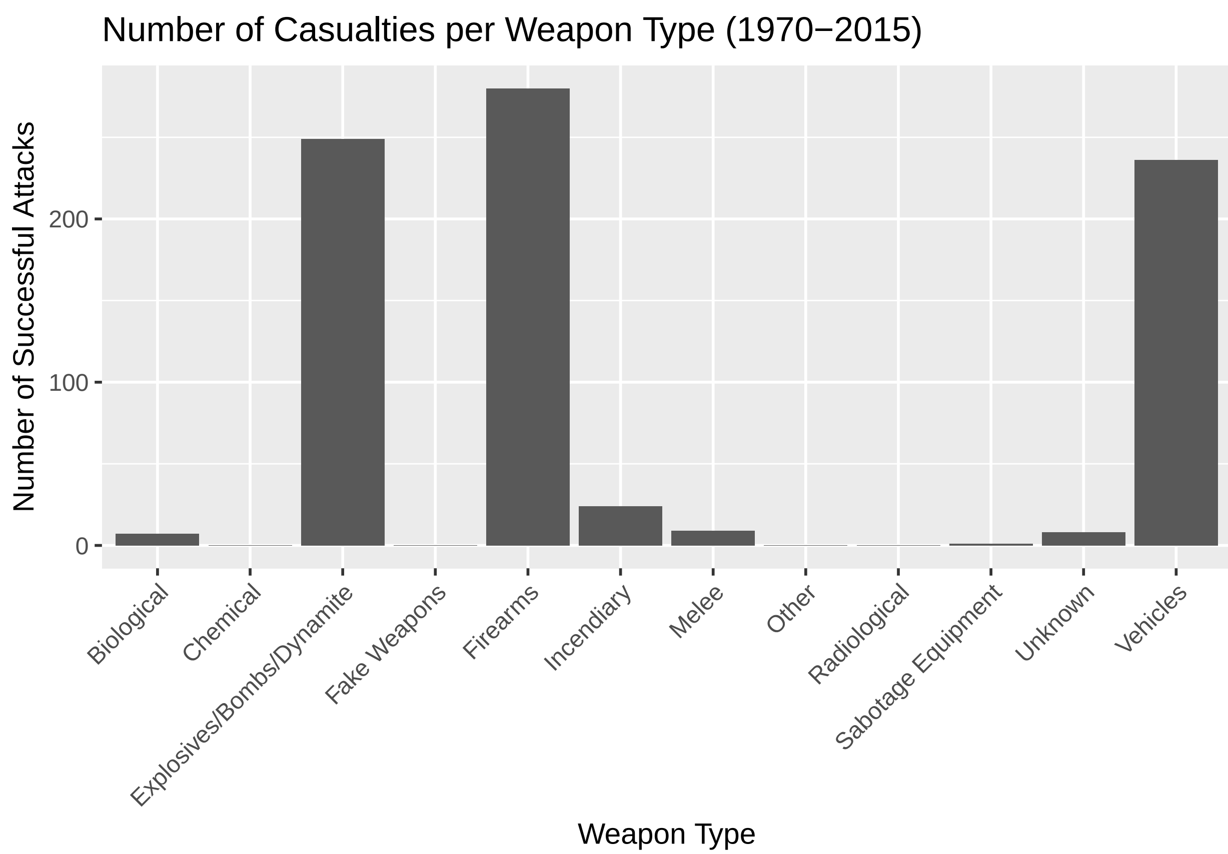


Finally, here's the number of attack successes by weapon type (again, no WTC):

```
# fix the long 'Vehicle' label with plyr
levels(gtd.noWTC$weaptype1_txt)
```

```
##  [1] "Biological"
##  [2] "Chemical"
##  [3] "Explosives/Bombs/Dynamite"
##  [4] "Fake Weapons"
##  [5] "Firearms"
##  [6] "Incendiary"
##  [7] "Melee"
##  [8] "Other"
##  [9] "Radiological"
## [10] "Sabotage Equipment"
## [11] "Unknown"
## [12] "Vehicle (not to include vehicle-borne explosives, i.e., car or truck bombs)"
```

```
gtd.noWTC$weaptype1_txt <- invisible(recode(gtd.noWTC$weaptype1_txt,
        "Vehicle (not to include vehicle-borne explosives, i.e., car or truck bombs)" = "Vehicles"))

# plot attack success by weapon type
gg.success.weap <- ggplot(gtd.noWTC, aes(weaptype1_txt, nkill))
gg.success.weap + geom_bar(stat="identity") + ylab("Number of Successful Attacks") +
  ggtitle("Number of Casualties per Weapon Type (1970-2015)") + xlab("Weapon Type") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1))
```

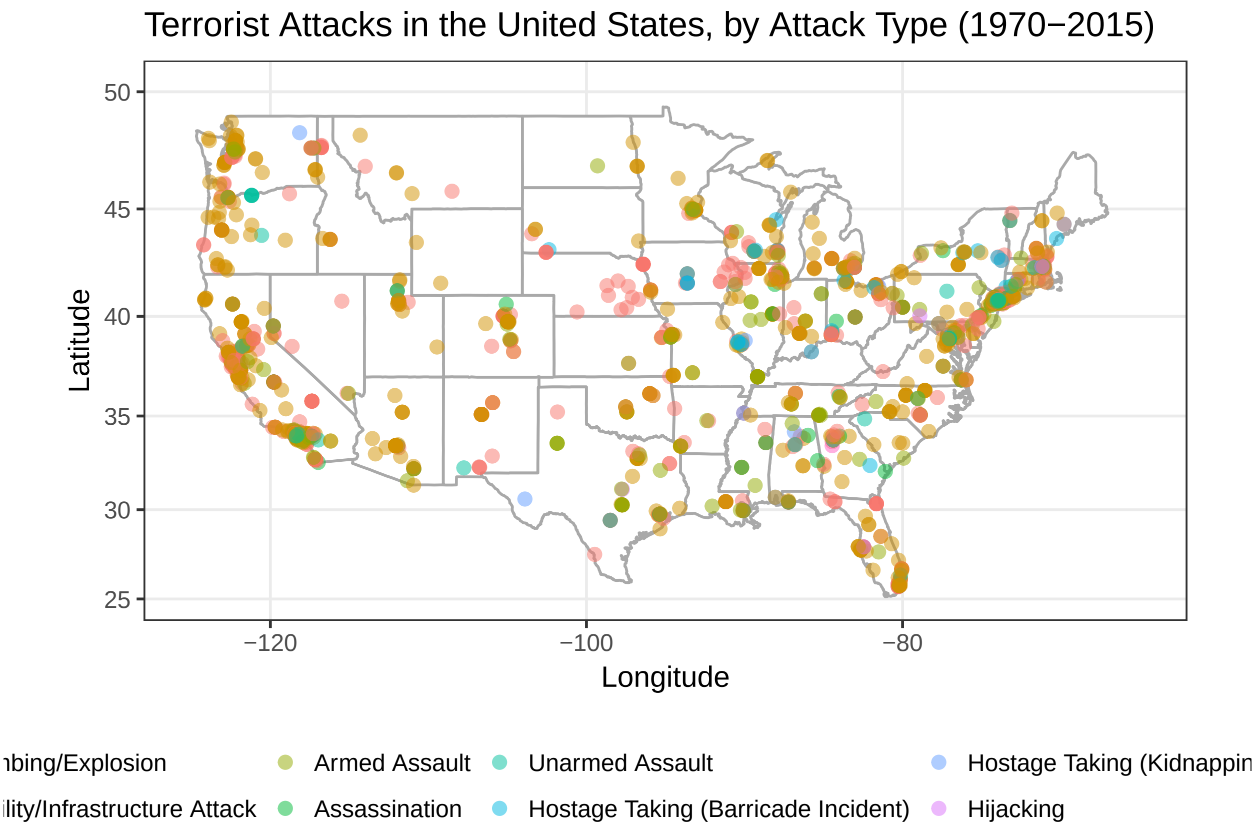


6) What are the other salient aspects of the data (e.g. geospatial factors, text content, etc.)

The data contains spatial coordinates for each attack, which we've mapped here for the 48 contigous United States:

```
library(maps)
us <- map_data("state")
q <- ggplot() +
  geom_polygon(data=us, aes(x=long, y=lat, group=group), color="darkgray", fill="white") +
  coord_map() + xlim(-125, -65) + ylim(25, 50) +
  geom_point(data=gtd, aes(x=longitude, y=latitude, color=attacktype1_txt), size=2, alpha=0.5) +
  theme_bw() + theme(legend.position="bottom") +
  ggtitle("Terrorist Attacks in the United States, by Attack Type (1970-2015)") +
  xlab("Longitude") + ylab("Latitude") +
  theme(legend.title=element_blank())
q
```

```
## Warning: Removed 257 rows containing missing values (geom_point).
```



The original dataset also included a description of each attack as a `summary` variable. While this is interesting, it is unrelated to our hypotheses and we've chosen to ignore it in this investigation.

7) Provide a bullet-list of the next 5-10 tasks you will perform in analyzing your dataset

What factors (target, attack type, weapons used etc.) predict whether or not an terrorist attack in the United States was a "success"?

1) Remove any confounding variables by testing for collinearity and determine what variables best predict whether or not a terrorist attack in the US was a "success".
2) Create an a-priori multi-variate logistic regression model to determine whether or not a terrorist attack in the US was a success.
3) Test our hypothesized model as well as variations that might improve its adjusted r-squared value (using stepwise regression).
4) Perform a k-fold validation on the model.

What factors (number of attackers, target, attack type, etc.), predict the number of casualties (victims killed or wounded)?

4) Remove any confounding variables by testing for collinearity and determine what variables best predict the number of casualties in an attack.
5) Create an a-priori multi-variate regression model, possibly a linear regression, best predict the number of casualties in an attack..

6) Test our hypothesized model as well as variations that might improve its adjusted r-squared value (using stepwise regression).
7) Perform a k-fold validation on the model.

What factors (motivation, terrorist organization type, attack type, etc.), predict whether or not a hostage or hostages were taken?

8) Remove any confounding variables by testing for collinearity and determine what variables best predict whether or not a hostage or hostages were taken.
9) Create an a-priori multi-variate logistic regression model to determine whether or not a hostage or hostages were taken.
10) Test our hypothesized model as well as variations that might improve its adjusted r-squared value (using stepwise regression).
11) Perform a k-fold validation on the model.

12) Plot a ROC curve and calculate AUC for all relevant models.
13) Use ColorBrewer2 to illustrate attacks.