Student Name: Homoud Alghanim

Student ID: 2132131306

Result:

1.

Variables identification

Column Name	Variable definition	Data Type	Missing data report	distribution of the data	level of analysis
eventid	The incidents ID from the Global Terrorism Database	Discrete(int64)	False	smallest first event id 197000000001 and the last 201701270001	Event
year	The year which the incident accured	Continues(int64)	False	Left Skewed	Event
month	The month which the incident accured	Continues(int64)	False	Left Skewed	Event
day	The Day which the incident accured	Continues(int64)	False	Right Skewed	Event
extended	The duration of an incident if it extended more than 24h or not	String	False	No Distribution	Event
summary	An information's summary of the incident	String	True(66138)	No Distribution	Event
Criterion_1	If the goal was for those things(political, economic, religious, or social goal) if it's 0 then was for something else.	String	False	No Distribution	Event

Column Name	Variable definition	Data Type	Missing data report	distribution of the data	level of analysis
Criterion_2	If the goal was for those things(intention to coerce, intimidate, or publicize to larger audiences) if it's 0 then was for something else.	String	False	No Distribution	Event
Criterion_3	If then the goal was something outside the international humanitarian law	String	False	No Distribution	Event
doubt_terr	If there's a doubt whether the incident is an act of terrorism.	String	False	No Distribution	Event
multiple	If the attack is part of a multiple incident	String	False	No Distribution	Event
related	The event ID related to this event if it was from multiple attacks	String	True(147505)	No Distribution	Event
city	The City where the incident occurred	String	True(446)	No Distribution	City
country	The Country where the incident occurred	String	False	No Distribution	Country
region	The Region where the incident occurred	String	False	No Distribution	Region
attacktype_1	The method of attack happened in the event	String	False	No Distribution	Event
attacktype_2	The method of attack happened in the event	String	True(164720)	No Distribution	Event
success	If the strike hasn't been stopped before it happened then it count as success	String	False	No Distribution	Event
weaptype	The type of weapon used in the event	String	False	No Distribution	Event
targ_type	The main target the terrorist want to strike	String	False	No Distribution	Event
targ_subtype	The sub target the terrorist want to strike	String	True(9345)	No Distribution	Event
nationality	The nationality of the terrorist or group that occured the event	String	True(1394)	No Distribution	Country
group_name	The name of group whom responsible for the event	String	False	No Distribution	Group

Column Name	Variable definition	Data Type	Missing data report	distribution of the data	level of analysis
kills_num	The number of people who died in the event	Continues(int64)	True(9682)	minimum kills 0 and maximum 1500	Event
wounded_num	The number of people who got wounded in the event	Continues(int64)	True(15325)	min 0 wounded maximum 7366	Event
latitude	The latitude of the city the event occurred in	Continues(float)	True(4606)	Left Skewed	Event
longitude	The longitude of the city the event occurred in	Continues(float)	True(4606)	Left Skewed	Event

2.

Transformed Variables

Variable Names	Variable description	Steps in transformation	level of analysis
crit_all'	It's an easy approuch for the thee column's(Criterion_1, Criterion_2, Criterion_3) joind them in one place	By applying the val function to go through multiple conditions	Event
top5_attacked_region	Has the information for the top 5 attacked regions	By getting the head of the value_counts and using isin while adding them	Region
top5_attacked_countries	The most 5 attacked countries informations	By getting the head of the value_counts and using isin while adding them	Country
top5_attacked_cities	The most 5 attacked cities informations	By getting the head of the value_counts and using isin while adding them	City
top5_target_nationality	The most 5 targeted nationalities informations	By getting the head of the value_counts and using isin while adding them	Country
top5_attacktype	The top 5 used attack types	By getting the head of the value_counts and using isin while adding them	Event
top5_target_type	The top 5 targeted types of people	By getting the head of the value_counts and using isin while adding them	Event
top5_target_Subtype	The top 5 Sub-targeted types of people	By getting the head of the value_counts and using isin while adding them	Event

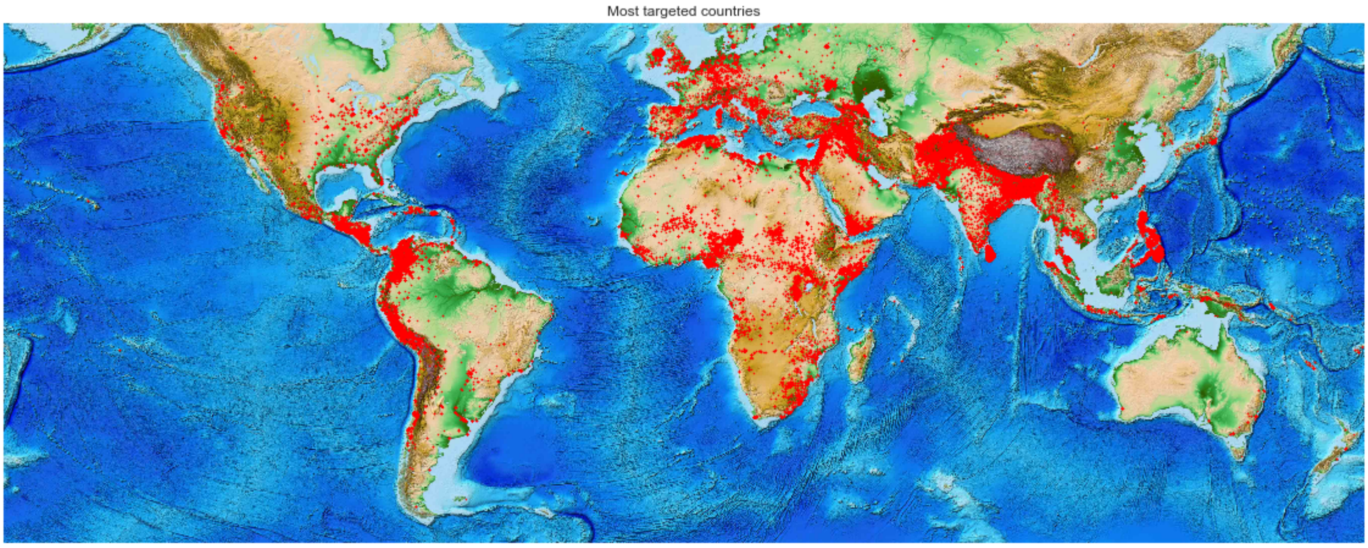
Variable Names	Variable description	Steps in transformation	level of analysis
top5_target_grpName	The top 5 active group named responsible for the event's	By getting the head of the value_counts and using isin while adding them	Group

3.

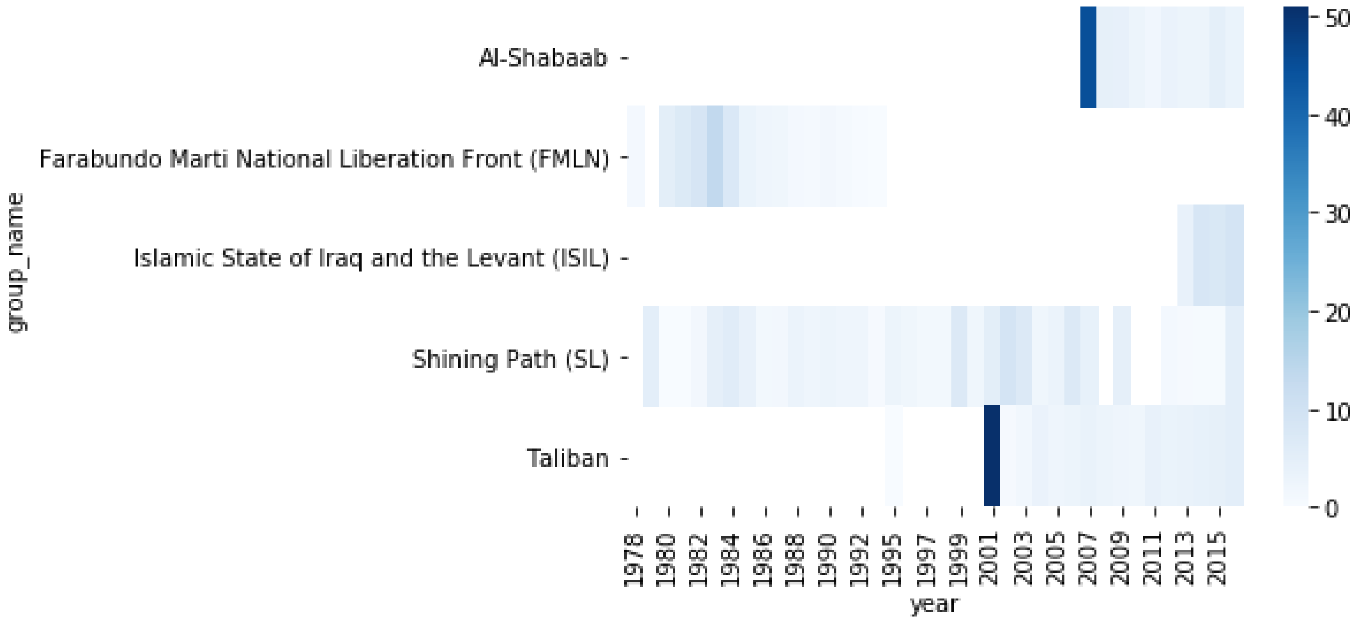
insights:

insight list:
Many event's before 1997 was not recoded if it was a terrorist act or not
May, Jul and November have many holidays in it or near it and the terrorist looks for clustered people, that's why they has a large amount of kills
The amount of people killed in the remaining months is kind of stable
Most people died from the terrorist attacks was in 2014
There is a decrease in the number of wounded people by the attacks after the year 2013
There is a decrease in the number of deads people by the attacks after the year 2014
ISIL now known as ISIS

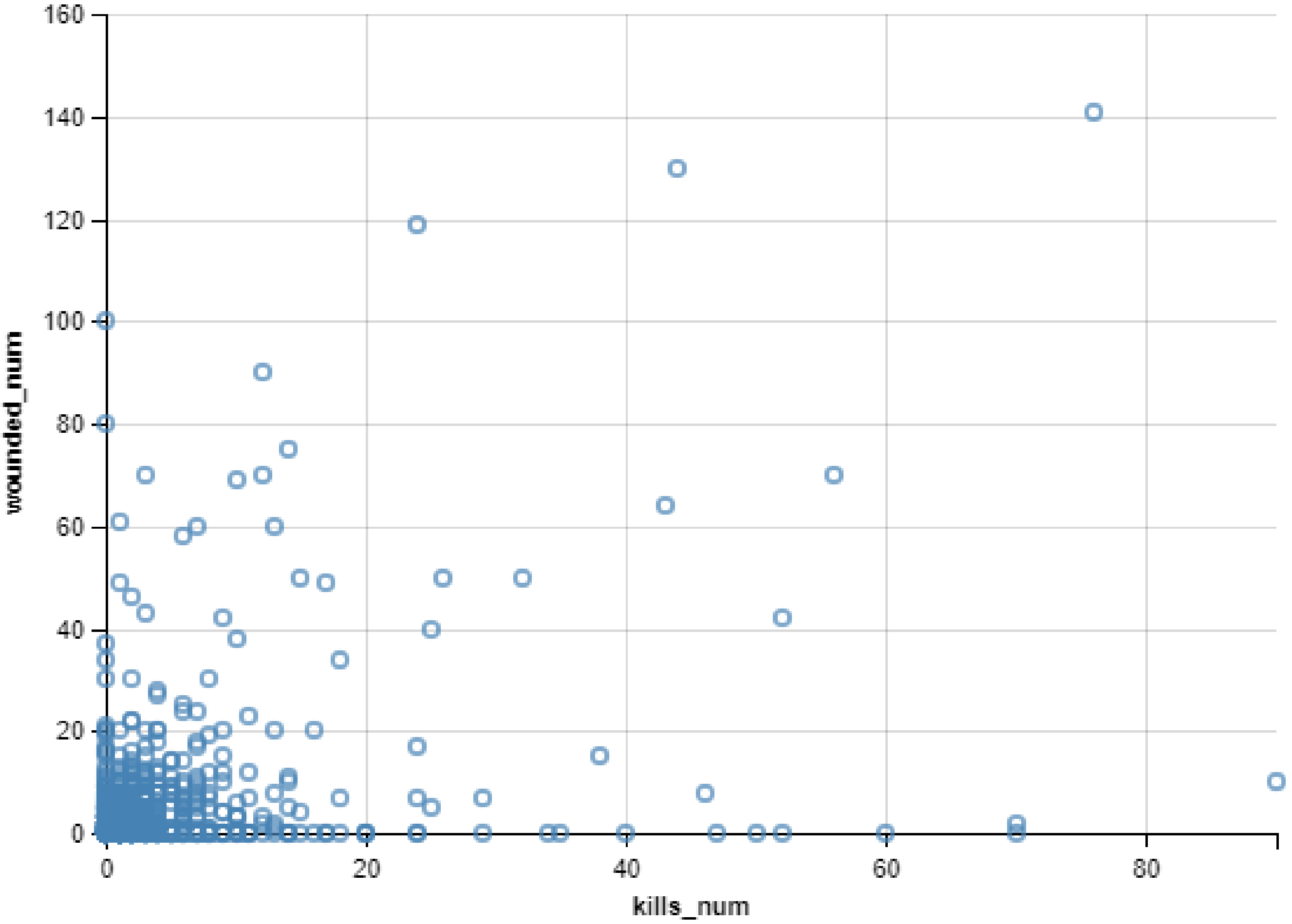
World plot of the most countries effected from terrorist



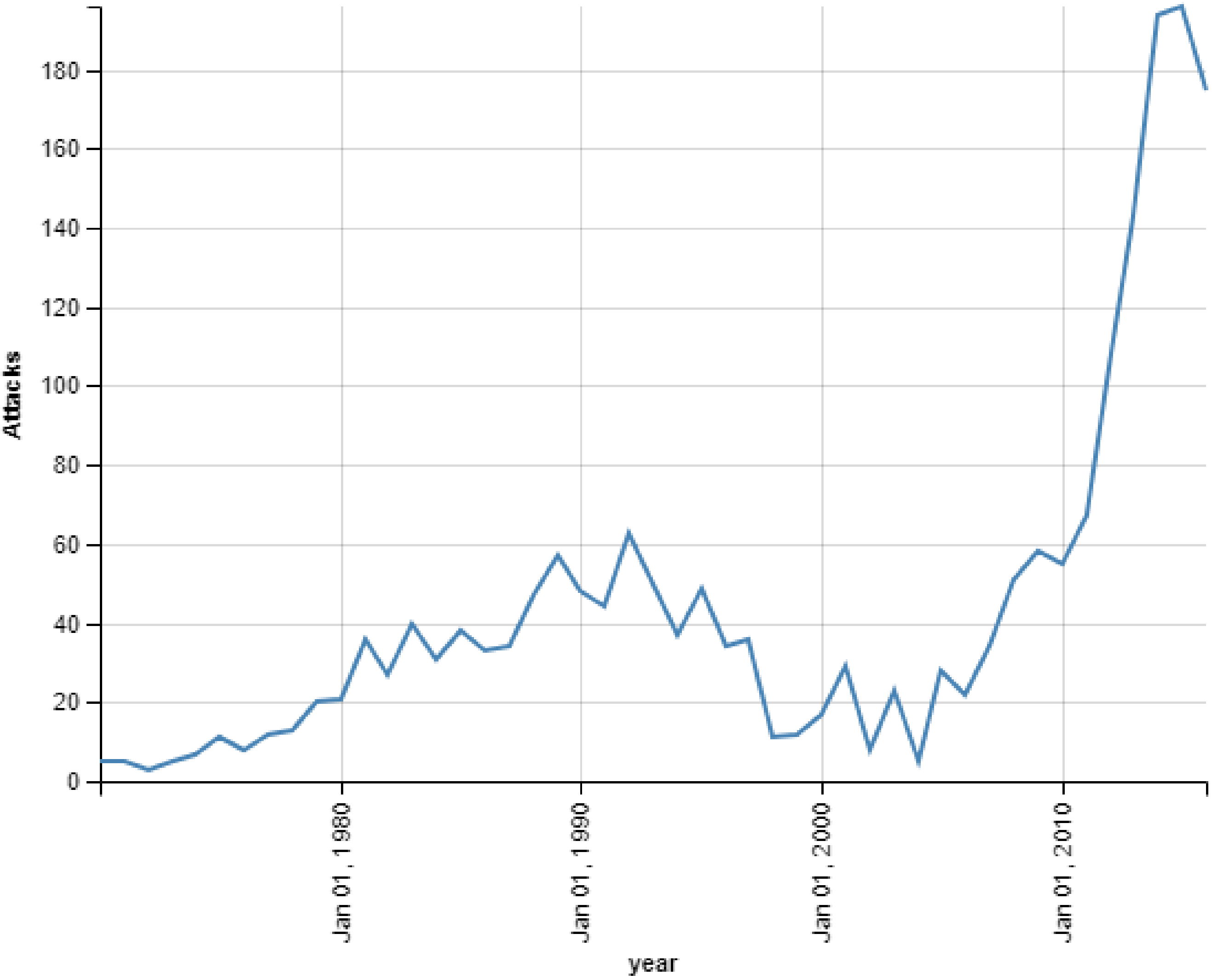
Why the numbers of killed people in 2001 increased very much from Taliban and from Al-Shabaab in 2007?



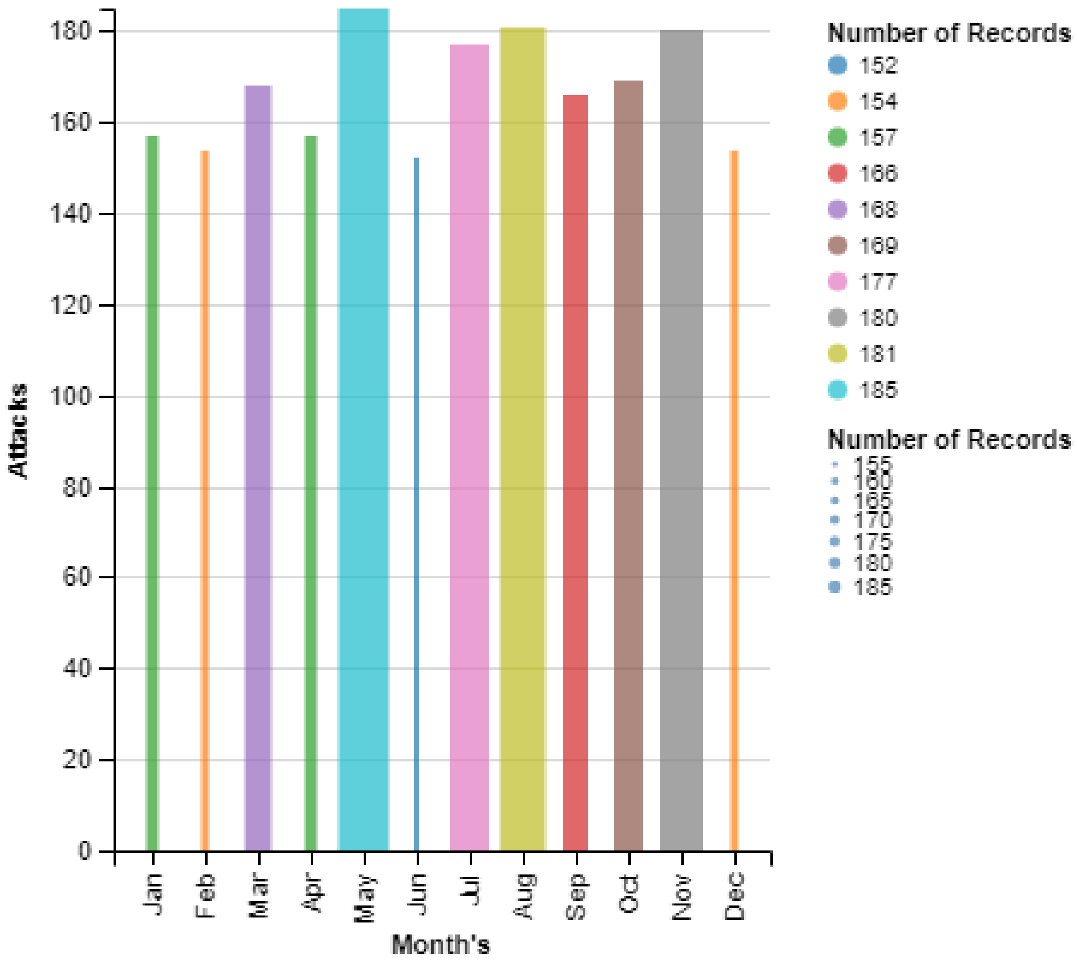
The wounded numbers increase every time the kills increase



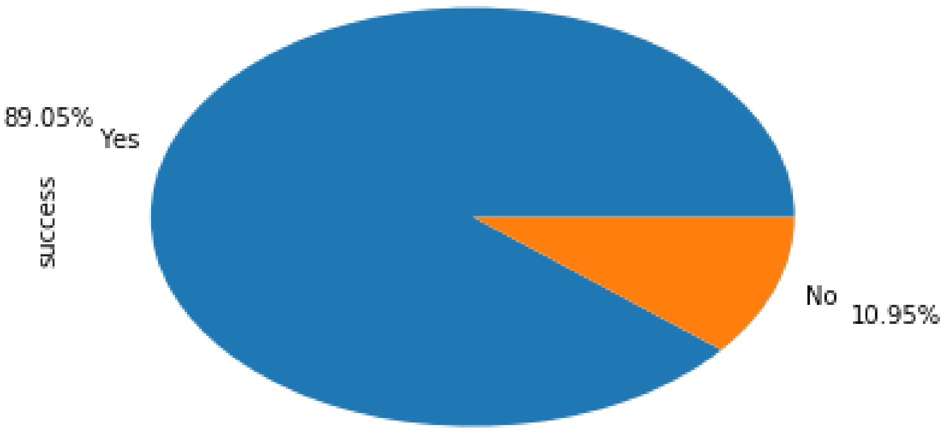
from the graph we can see that 2014 was the highest



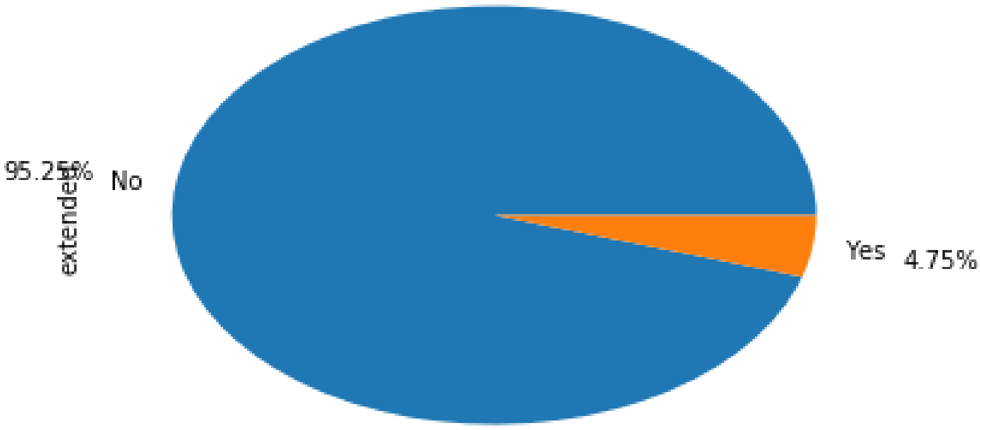
The graph shows that May, August and november are the highest, maybe because of the holidays?



89.05% the operations of the terrorist succeeded



less than 5% of the strikes in the database, event's stayed more than 24 hour

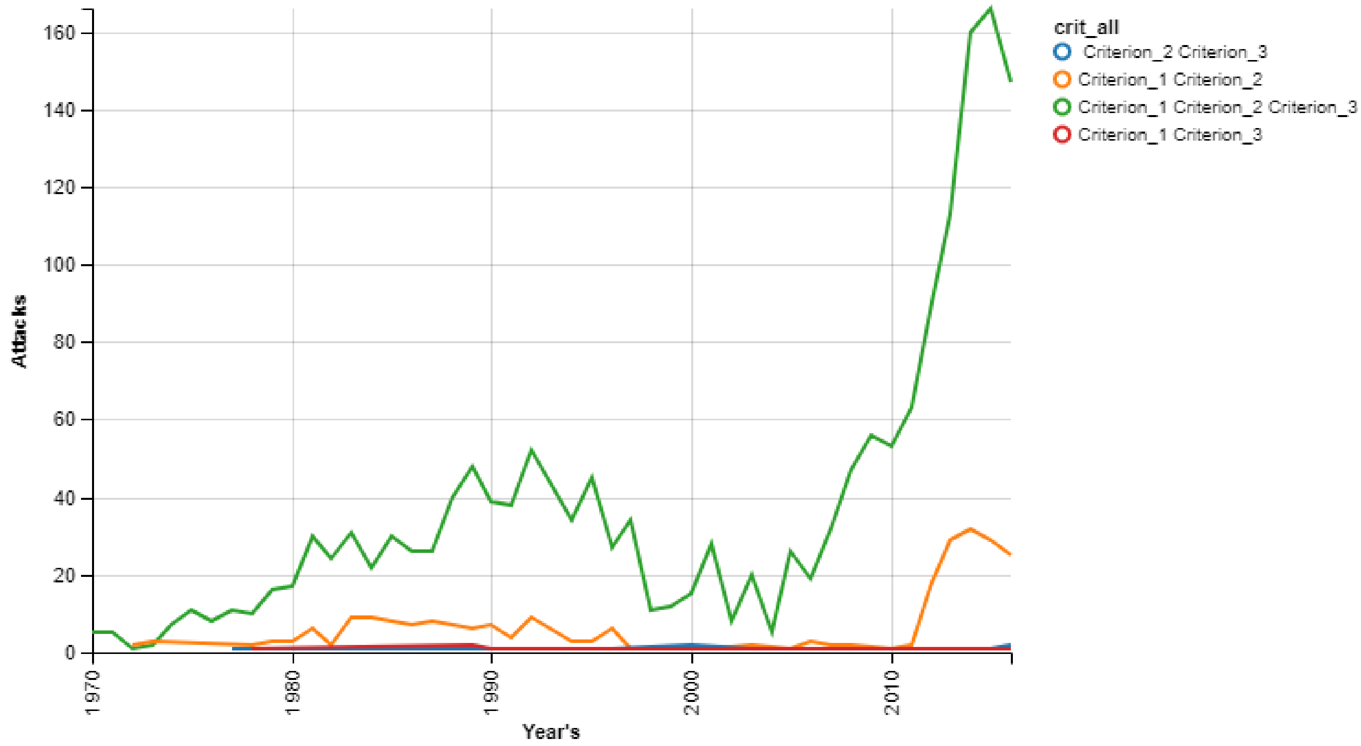


We can see that most attacks choose all the three:

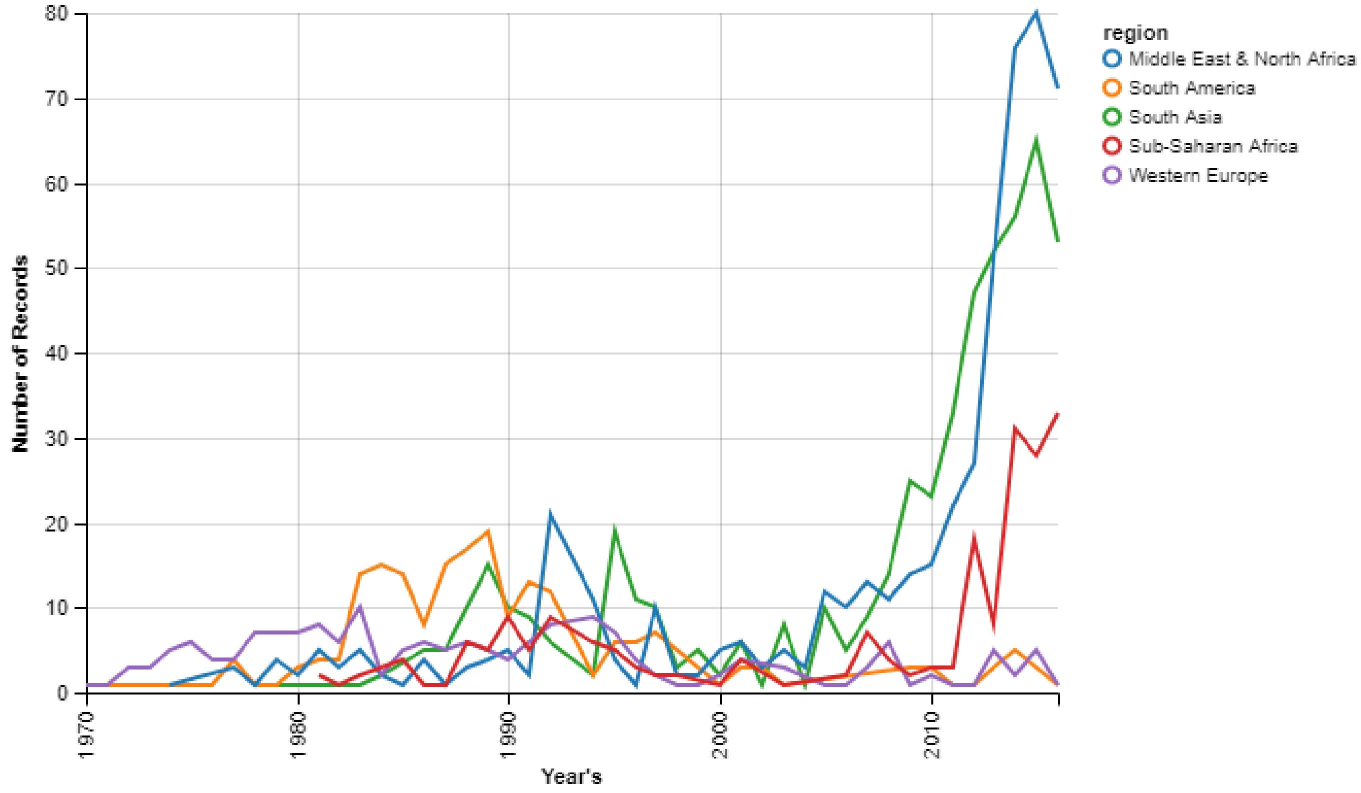
Criterion_1 (political, economic, religious, or social goal),

Criterion_2 (intention to coerce, intimidate, or publicize to larger audiences)

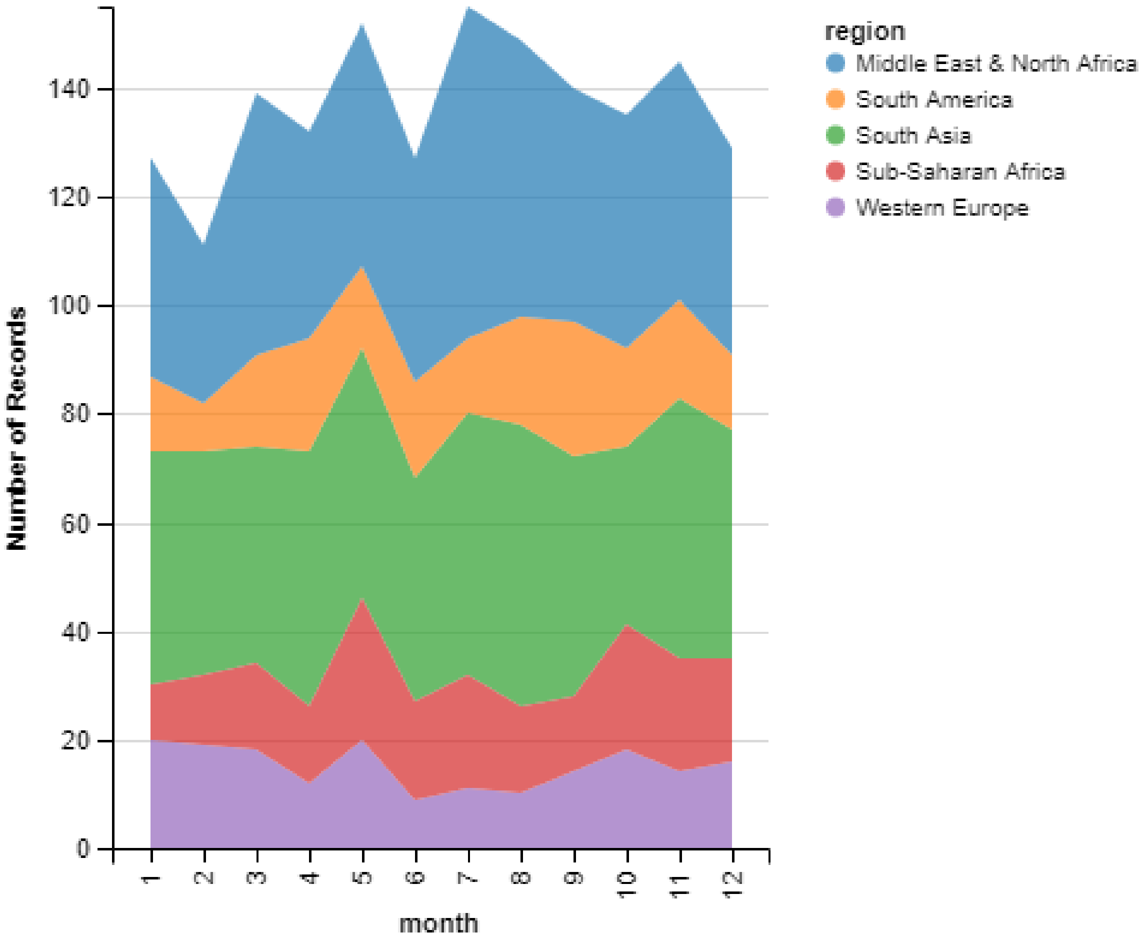
Criterion_3 Something outside the international humanitarian law



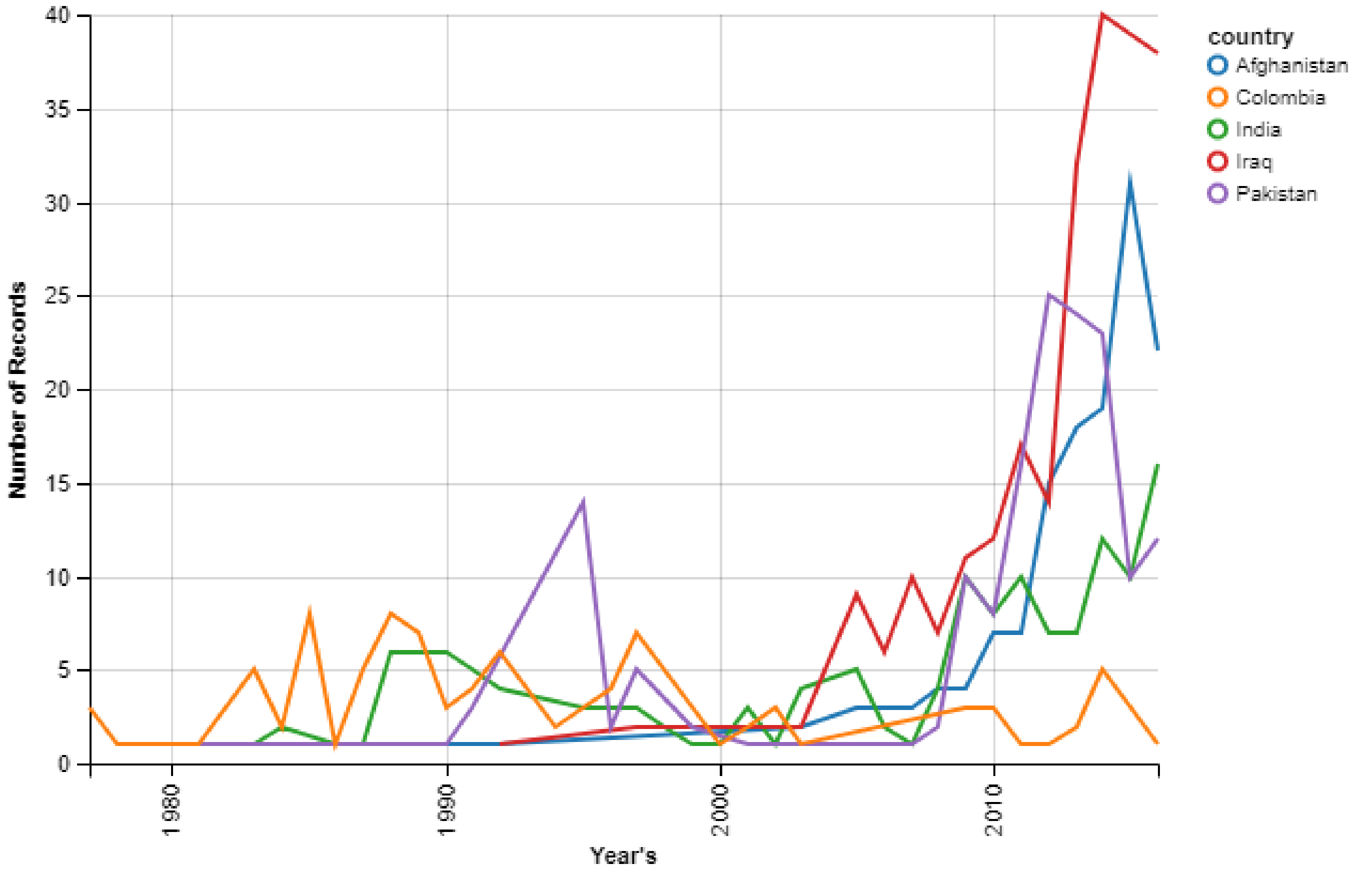
The Middle East and North Africa are the most attacked region



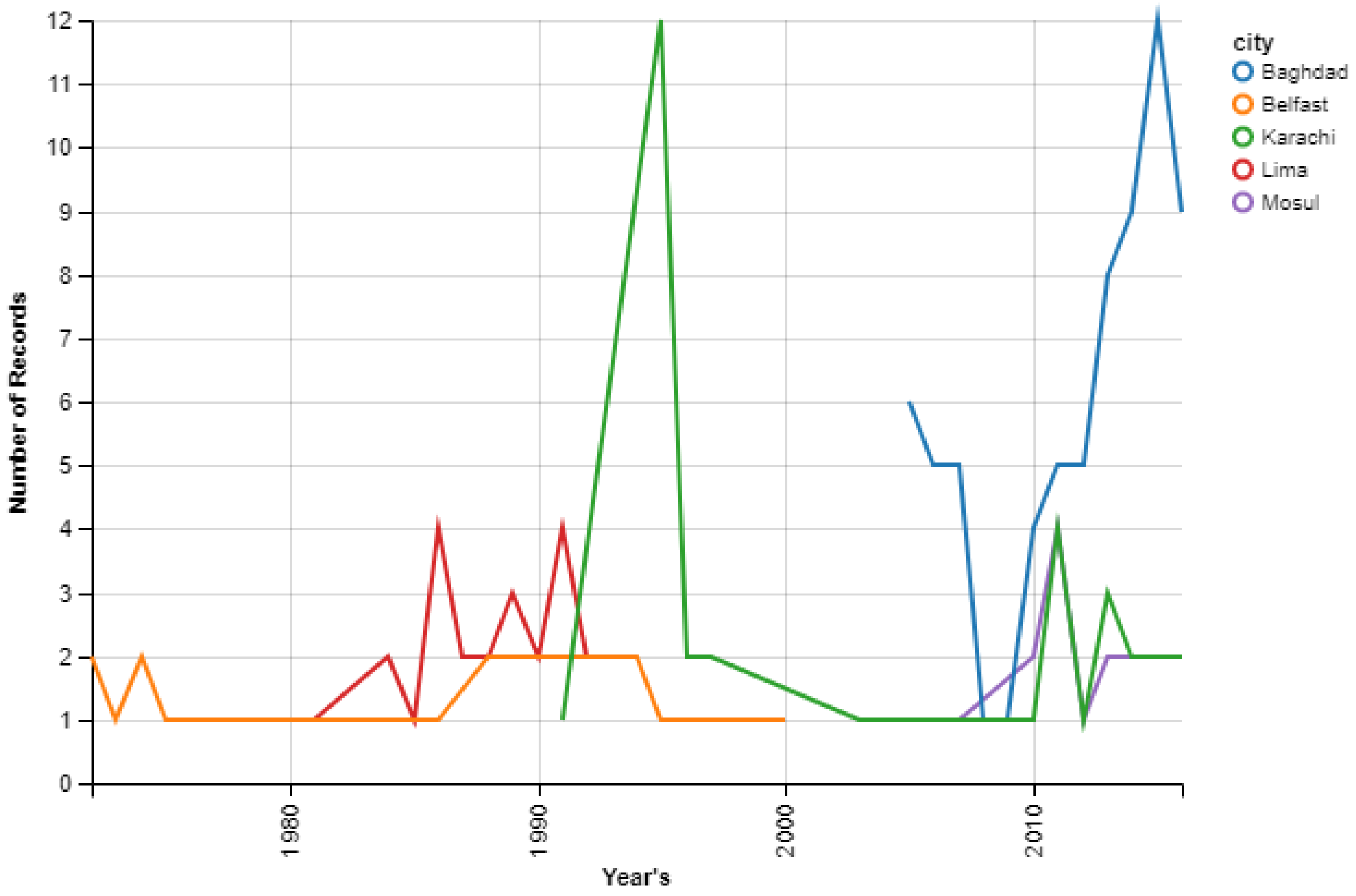
The most active month in the Middle east & North Africa is july



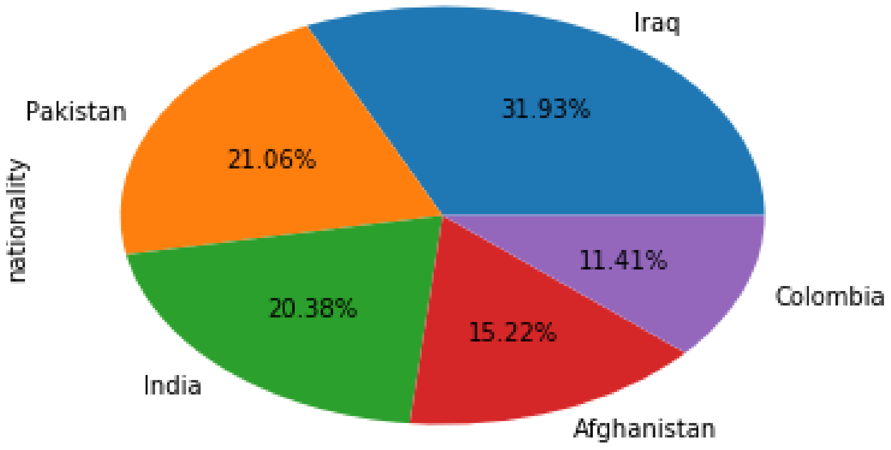
Iraq attacks increased after saddam hussain death in 2003



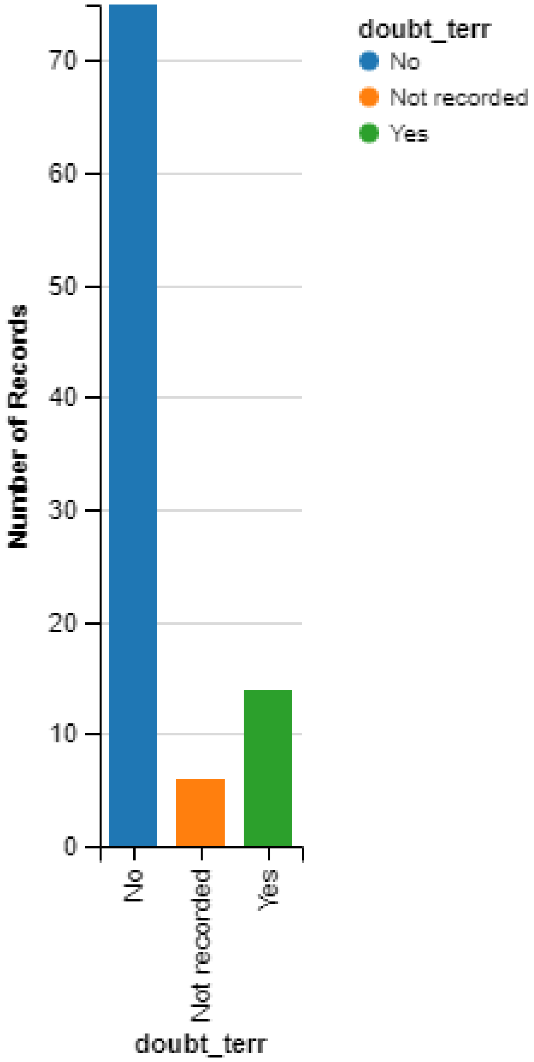
Baghdad attacks spiked after 1991 and Lima city attacks stopped in 1992



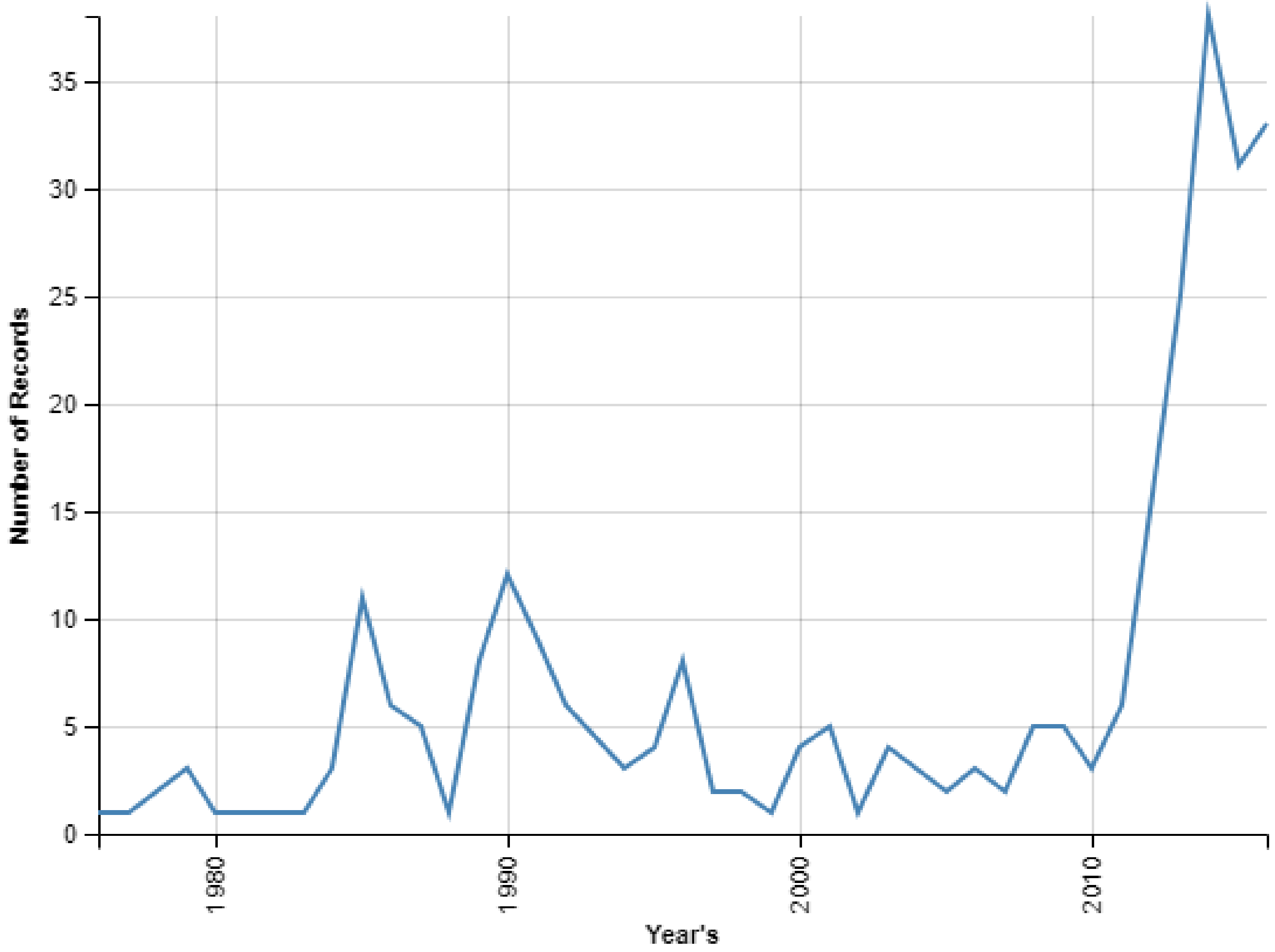
The percentage of the attacks on those nationality



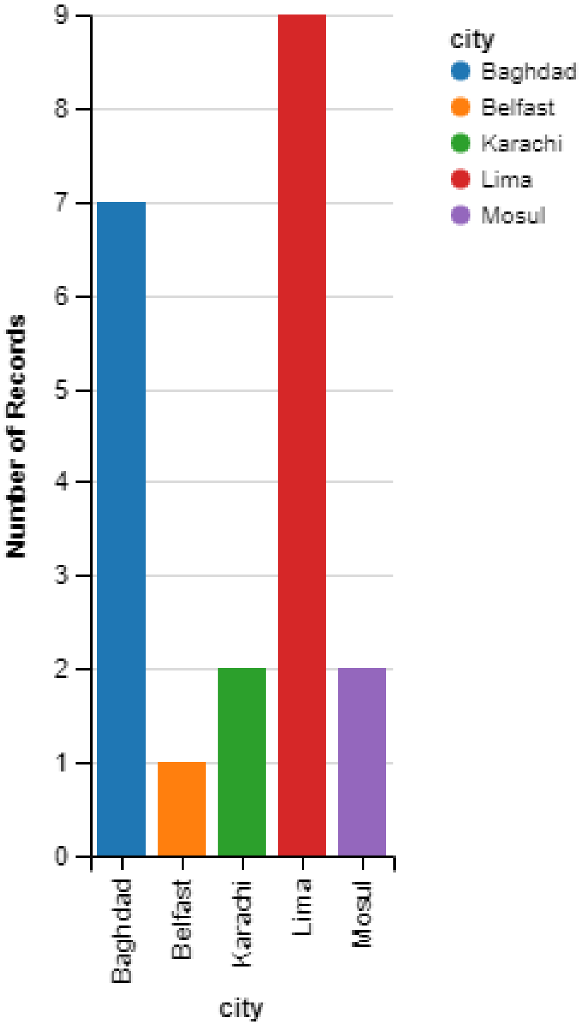
Even after the event's extended more than 24 hour there's small amount still doubting if it's a terrorism attack or not



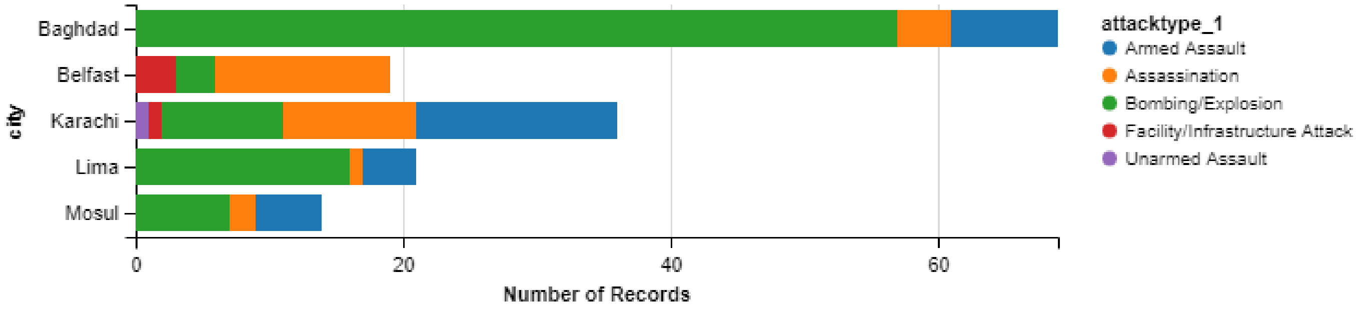
The event which is part of multiple event's spiked after 2011



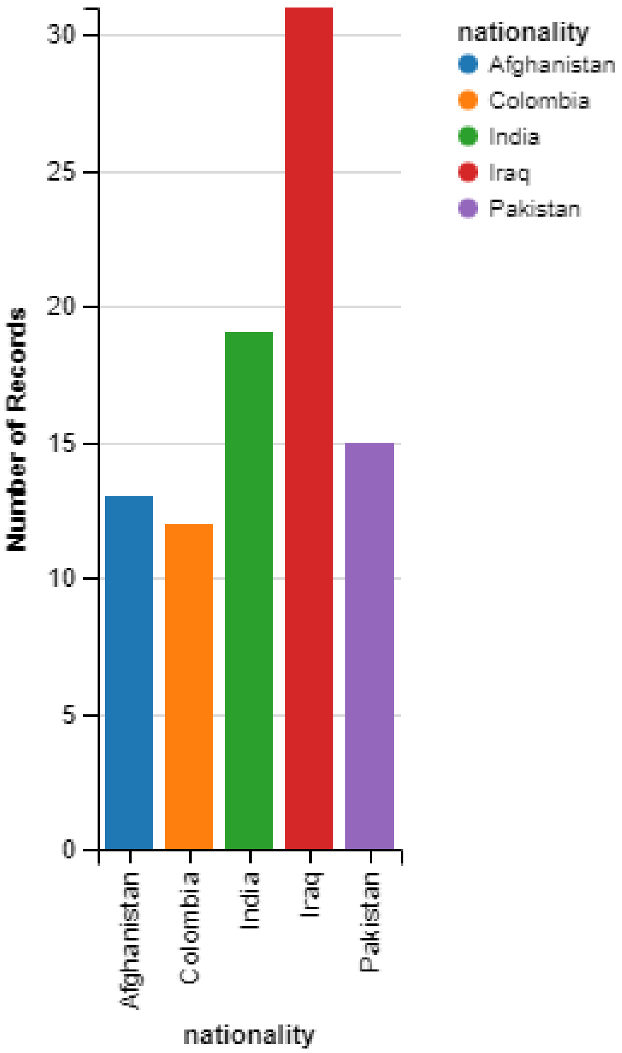
Lima and Baghdad got the most multiple event's



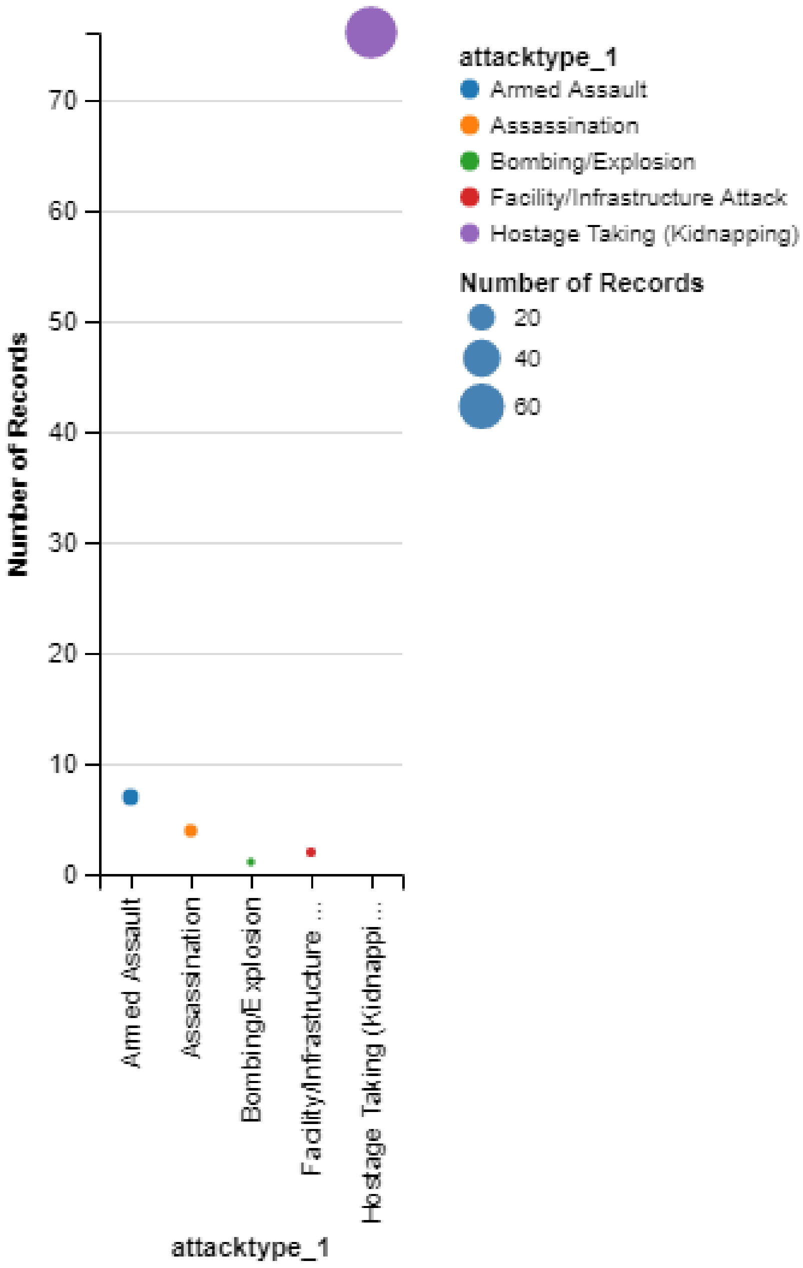
Baghdad got bombed the most then Karachi got Armed assault the most



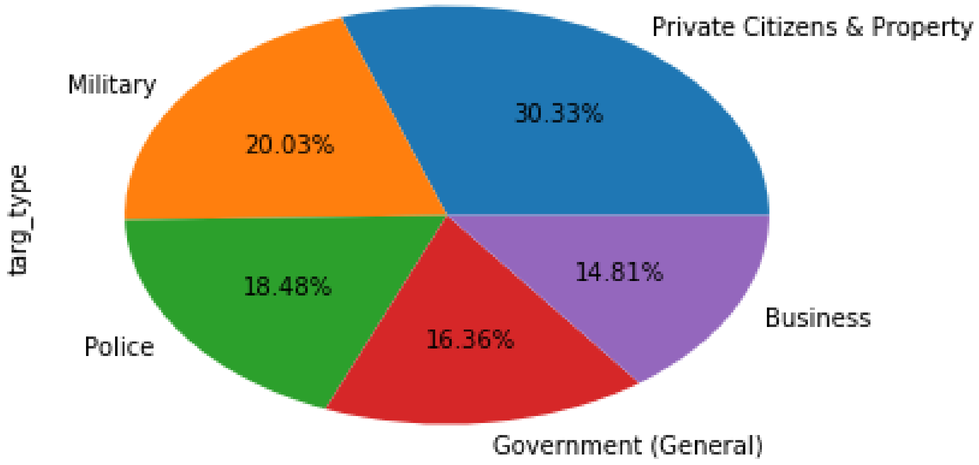
The highest nationalities who got multiple event's are Iraqis and India



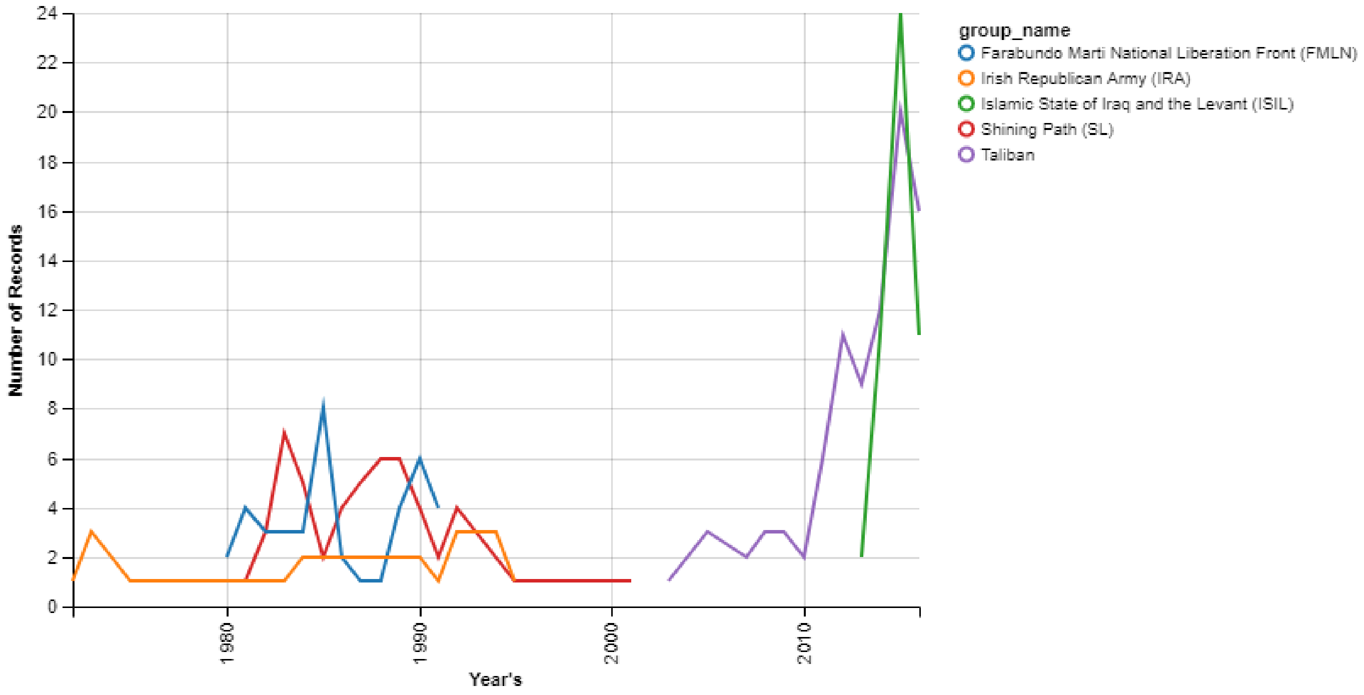
The Kidnapping attack type looks like the most frequent one used in the event's that extends for more than 24 hour's



most attacks target type are Private Citizens & Property



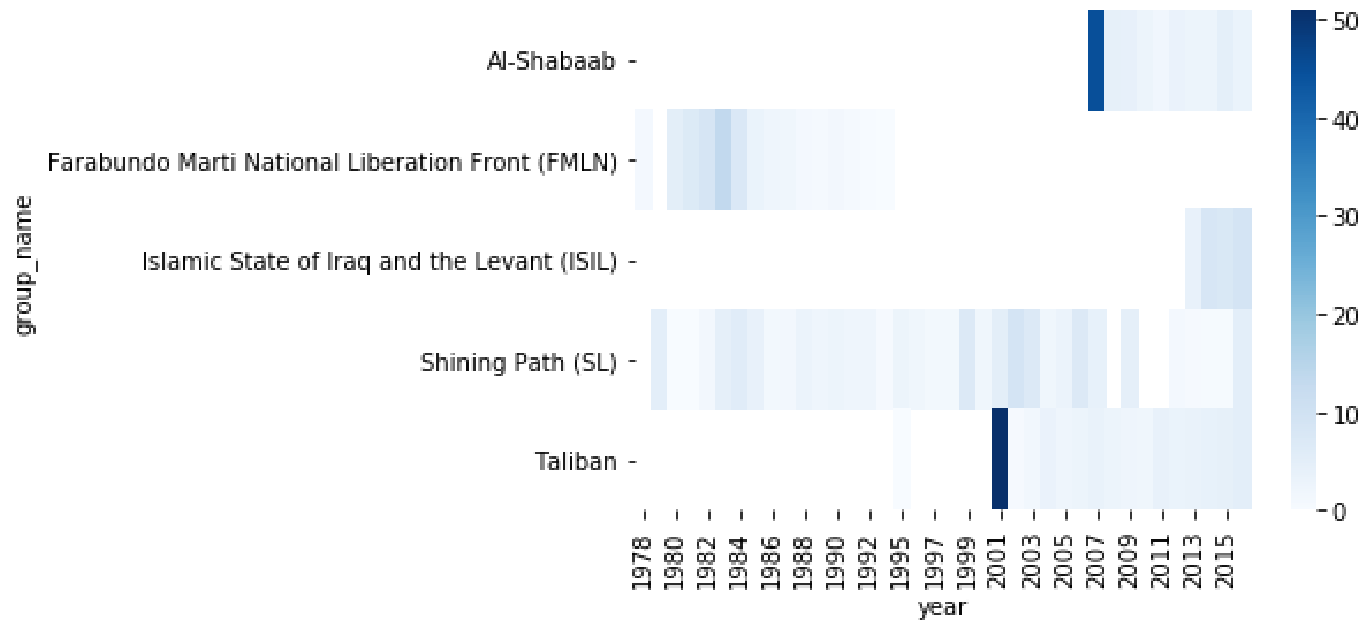
we can see that the ISIL and Taliban the most active groups



4.

Question's:

- Why all the terrorist activities increased mostly after 2010?
- Can we predict about when the next attack if we feed this information to a machine?
- The countries with the less damage and attacks are they good at defending themselves or they're not targeted?
- Why the numbers of killed people in 2001 increased very much from Taliban and from Al-Shabaab in 2007?



Analysis:

In [1]:

```
import pandas as pd
import altair as alt
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

%matplotlib inline
```

In [2]:

```
df=pd.read_csv('Global_Terrorism_Database.csv',low_memory=False)
pd.options.display.max_columns=False
```

In [ ]:

```

```

In [12]:

```
df.head()
```

Out[12]:

	eventid	year	month	day	extended	summary	Criterion_1	Crit
0	198808180003	1988	8	18	No	NaN	Yes	
1	198301030001	1983	1	3	No	NaN	Yes	
2	201208060015	2012	8	6	No	08/06/2012: Three gunmen armed with automatic ...	Yes	
3	201507250018	2015	7	25	No	07/25/2015: Assailants attacked an army base i...	Yes	
4	201304250031	2013	4	25	No	04/25/2013: Assailants threw a petrol bomb at ...	Yes	

In [6]:

```
df.describe()
```

Out[6]:

	eventid	year	month	day	ext
count	1.703500e+05	170350.000000	170350.000000	170350.000000	170350.0
mean	2.001776e+11	2001.709997	6.474365	15.466845	0.0
std	1.314444e+09	13.144146	3.392364	8.817929	0.2
min	1.970000e+11	1970.000000	0.000000	0.000000	0.0
25%	1.990053e+11	1990.000000	4.000000	8.000000	0.0
50%	2.007121e+11	2007.000000	6.000000	15.000000	0.0
75%	2.014023e+11	2014.000000	9.000000	23.000000	0.0
max	2.017013e+11	2016.000000	12.000000	31.000000	1.0

In [5]:

```
df.dtypes
```

Out[5]:

```
eventid            int64
year               int64
month              int64
day                int64
extended           int64
summary           object
Criterion_1        int64
Criterion_2        int64
Criterion_3        int64
doubt_terr         int64
multiple           int64
related           object
city              object
country           object
region            object
attacktype_1      object
attacktype_2      object
success            int64
weaptype          object
targ_type         object
targ_subtype      object
nationality       object
group_name        object
kills_num        float64
wounded_num      float64
latitude         float64
longitude        float64
dtype: object
```

In [6]:

```
print(df.isnull().sum())
#We can see the each column and how many missing data it has.
```

```
eventid              0
year                 0
month                0
day                  0
extended             0
summary          66138
Criterion_1          0
Criterion_2          0
Criterion_3          0
doubt_terr           0
multiple             0
related         147505
city               446
country              0
region               0
attacktype_1         0
attacktype_2    164720
success              0
weaptype             0
targ_type            0
targ_subtype      9345
nationality       1394
group_name           0
kills_num         9682
wounded_num      15325
latitude          4606
longitude         4606
dtype: int64
```

In [71]:

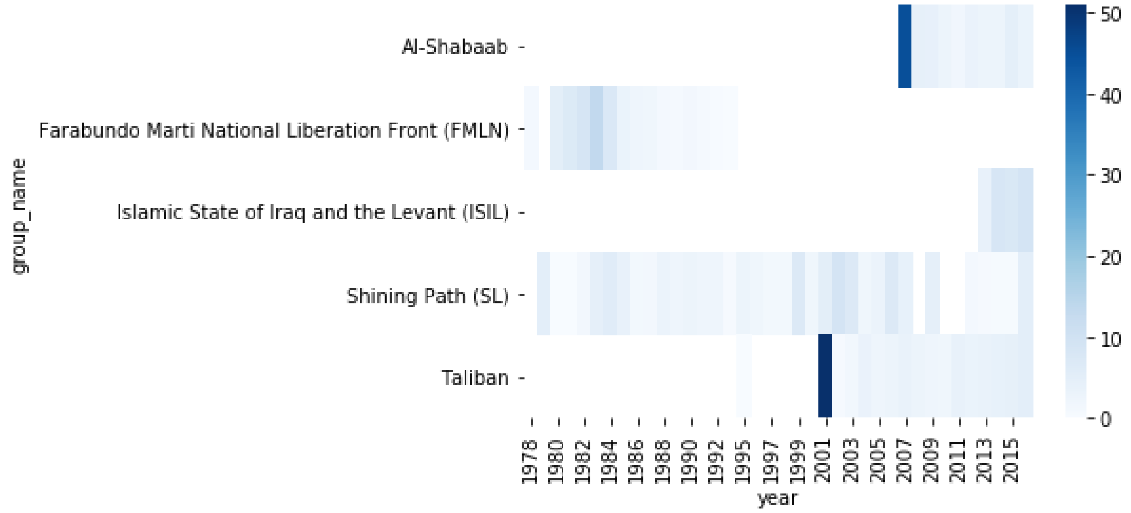
```
df.corr()
```

Out[71]:

```
                eventid      year     month       day  extended  Criteric
    eventid    1.000000  0.999996  0.006579  0.015797  0.090726     0.001
       year    0.999996  1.000000  0.003989  0.015718  0.090716     0.001
      month    0.006579  0.003989  1.000000  0.004970  0.000017     0.001
        day    0.015797  0.015718  0.004970  1.000000 -0.004475     0.011
   extended    0.090726  0.090716  0.000017 -0.004475  1.000000    -0.015
Criterion_1    0.001329  0.001304  0.001393  0.011779 -0.015347     1.000
Criterion_2    0.026638  0.026635  0.001366 -0.002234  0.001771    -0.009
Criterion_3   -0.003225 -0.003218 -0.000917 -0.004240  0.048633    -0.039
 doubt_terr    0.290603  0.290638 -0.012174  0.001049  0.010386    -0.059
   multiple    0.094126  0.094136 -0.002086 -0.005037 -0.014125     0.029
    success   -0.070257 -0.070249 -0.002810 -0.008578  0.069034    -0.009
  kills_num    0.015053  0.015042  0.004255 -0.003799  0.022931     0.010
wounded_num    0.018573  0.018565  0.002945 -0.000979 -0.007283     0.007
   latitude    0.171345  0.171396 -0.015723  0.002978 -0.021142     0.020
  longitude    0.553012  0.553031 -0.004085  0.010984  0.036189    -0.005
```

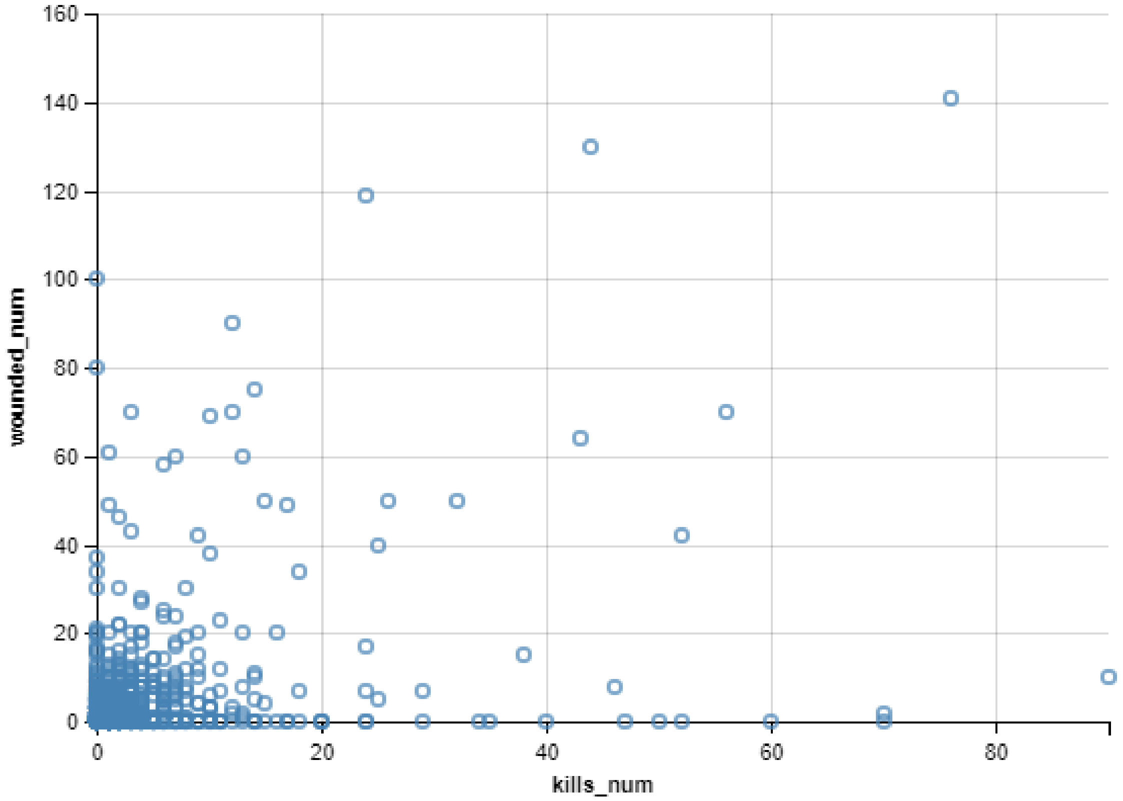
In [24]:

```
ax= sns.heatmap(
    df.pivot_table(
        'kills_num',
        top5_target_grpName.group_name[top5_target_grpName['group_name'].no
        'year'),cmap="Blues")

# Why the numbers of killed people in 2001 increased very much from Taliban
# Those data was taken directly from the dataframe not the sample
```



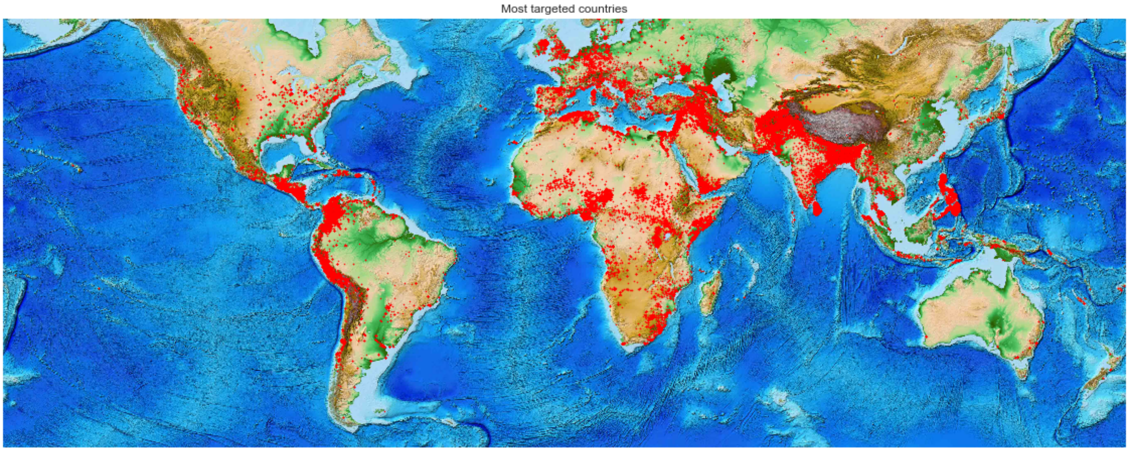
In [237]:

```
alt.Chart(df_sample[(df_sample.wounded_num.notnull())&(df_sample.kills_num.
    x='kills_num',
    y='wounded_num'
)

# The wounded numbers increase every time the kills increase
```



In [222]:

```
import matplotlib.pyplot as plt
from mpl_toolkits.basemap import Basemap

plt.figure(figsize=(20,16))
targeted_countries = df['country']

m = Basemap(projection='mill',llcrnrlat=-60,urcrnrlat=60,\
            llcrnrlon=-180,urcrnrlon=180,resolution=None)

x, y = m(list(df["longitude"].astype("float")), list(df["latitude"].astype(
m.plot(x, y,'.', markersize = 3, alpha = 1, color = "red")

m.etopo()
plt.title('Most targeted countries')
plt.show()

# World plot of the most countries effected from terrorist
```



In [3]:

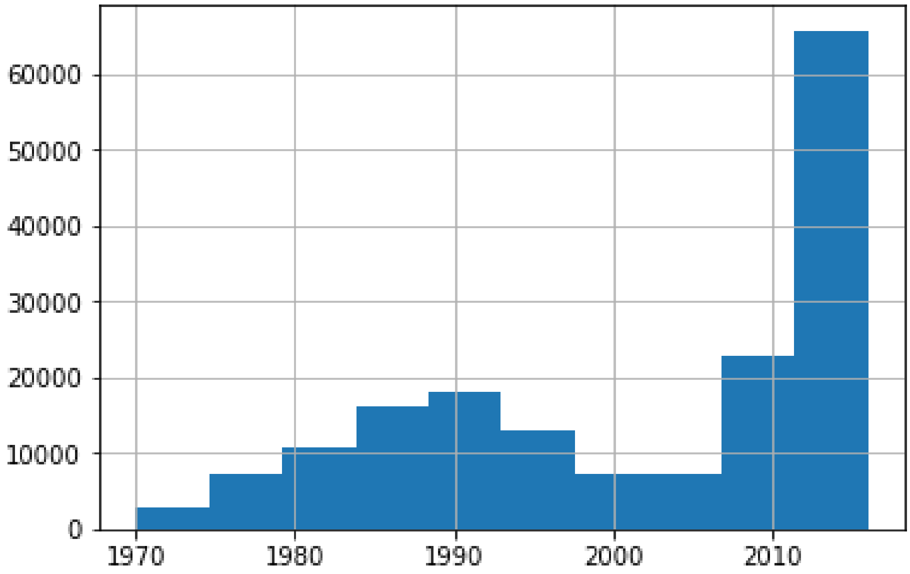
```
def con(x):

    if str(x)=='nan' or str(x)=='None':
        return x
    else:
        return int(np.round(x))

df['kills_num']=df['kills_num'].apply(con)
df['wounded_num']=df['wounded_num'].apply(con)
#Here we needed to change data type to integer because we can't kill or inj
```

In [8]:

```
df['year'].hist()
# Left skewed distribution
```

Out[8]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99a0be908>
```

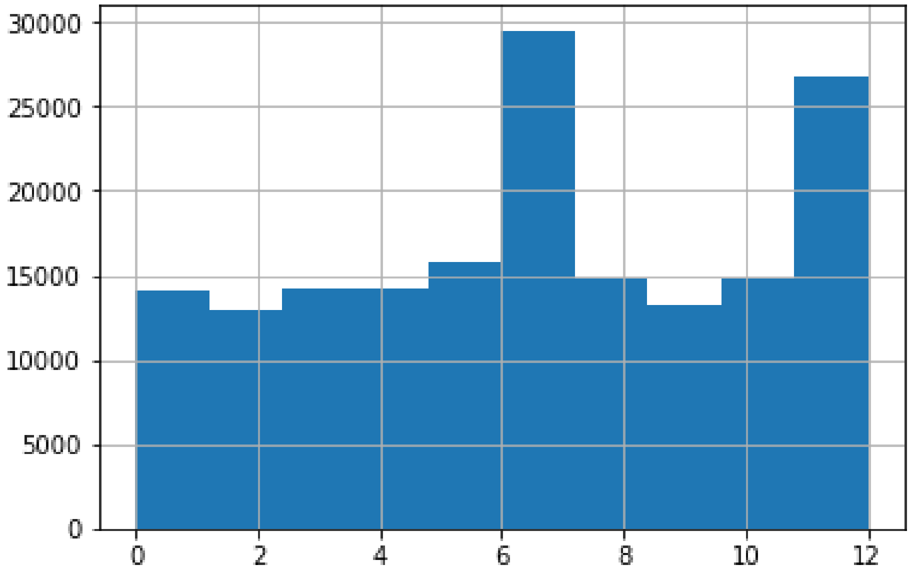


In [9]:

```
df['month'].hist()
# Left skewed distribution
```

Out[9]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99a5b8cf8>
```

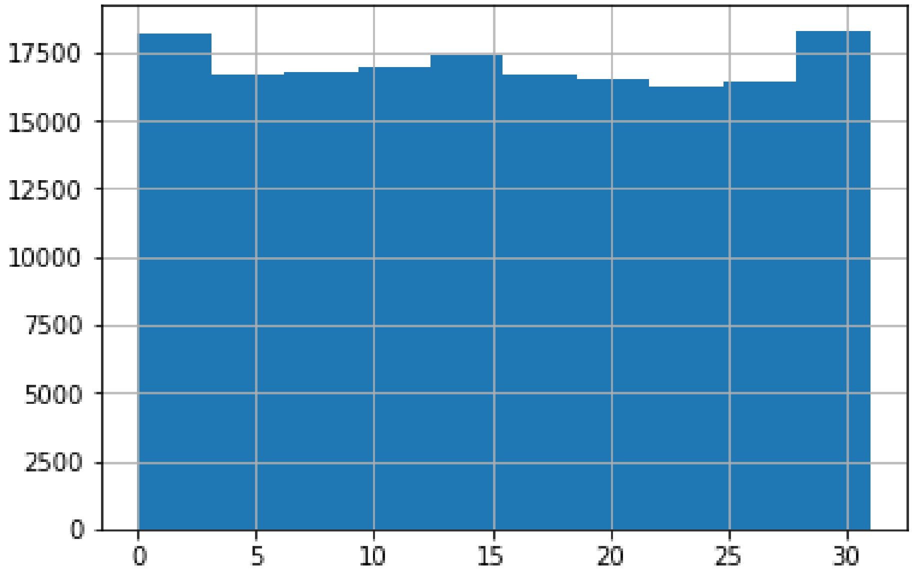


In [10]:

```
df['day'].hist()
# Right skewed distribution
```

Out[10]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99a160c88>
```

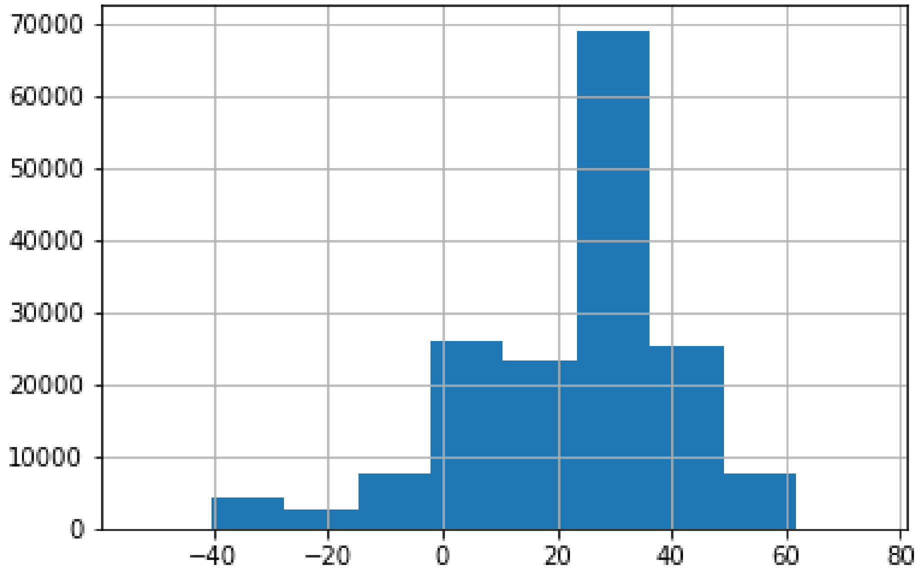


In [11]:

```
df['latitude'].hist()
#The distribution skewed to the left
```

Out[11]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99a201fd0>
```

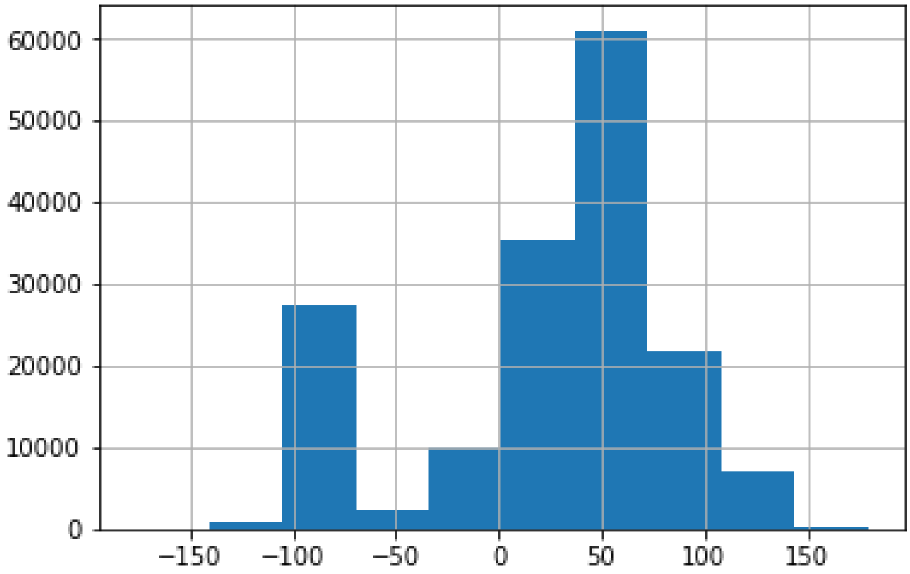


In [12]:

```
df['longitude'].hist()
#The distribution skewed to the left
```

Out[12]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99a2e20f0>
```



In [15]:

```
df_sample=df.sample(2000).reset_index(drop=True)
# Taking a sample from the dataframe because altasir can't take more than 5
```

In [4]:

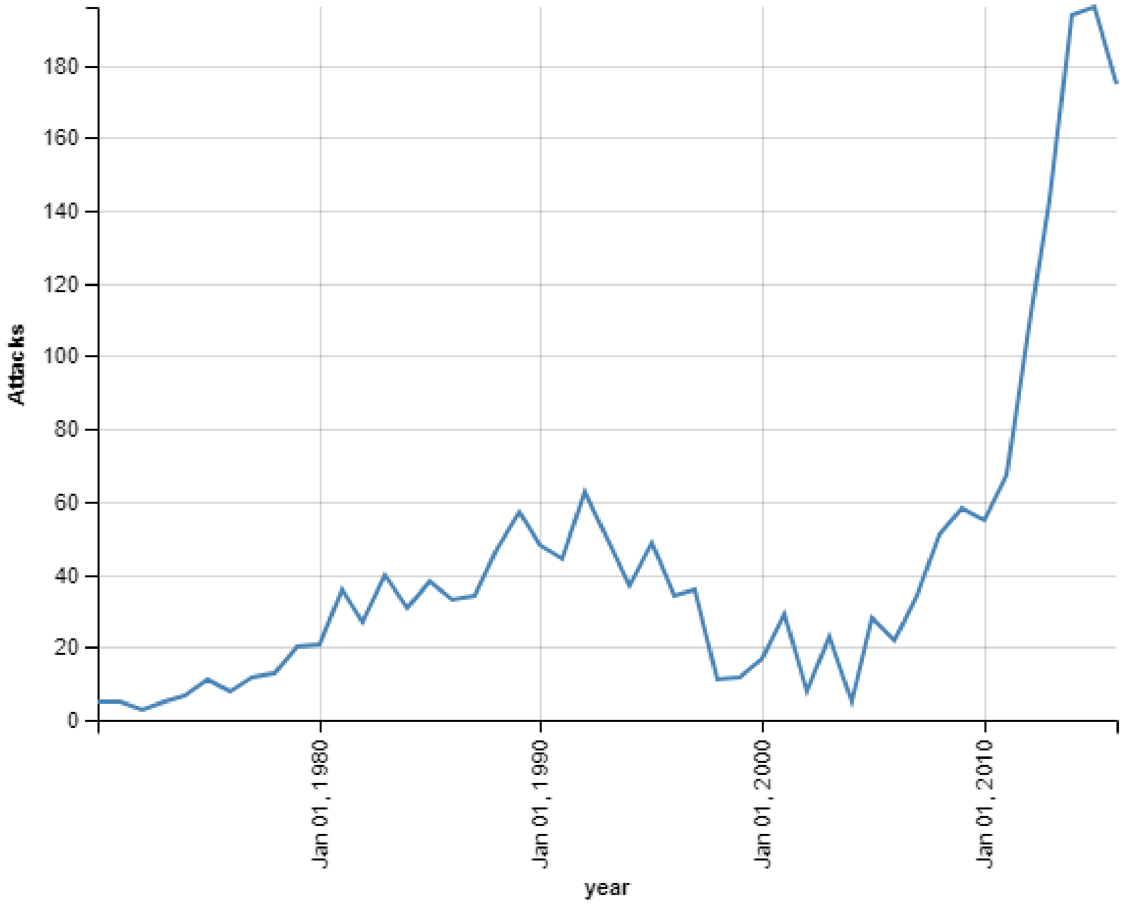
```
def val(x):

    if str(x)=='nan' or str(x)=='None':
        return x
    elif x==1:
        return 'Yes'
    elif x==-9:
        return 'Not recorded'
    else:
        return 'No'
```

In [6]:

```
df_sample['extended']=df_sample['extended'].apply(val)
df_sample['Criterion_1']=df_sample['Criterion_1'].apply(val)
df_sample['Criterion_2']=df_sample['Criterion_2'].apply(val)
df_sample['Criterion_3']=df_sample['Criterion_3'].apply(val)
df_sample['multiple']=df_sample['multiple'].apply(val)
df_sample['doubt_terr']=df_sample['doubt_terr'].apply(val)
df_sample['success']=df_sample['success'].apply(val)

# Changing the values from 1 to Yes and 0 to No
# So when we plot them they'll be easy to read.
```

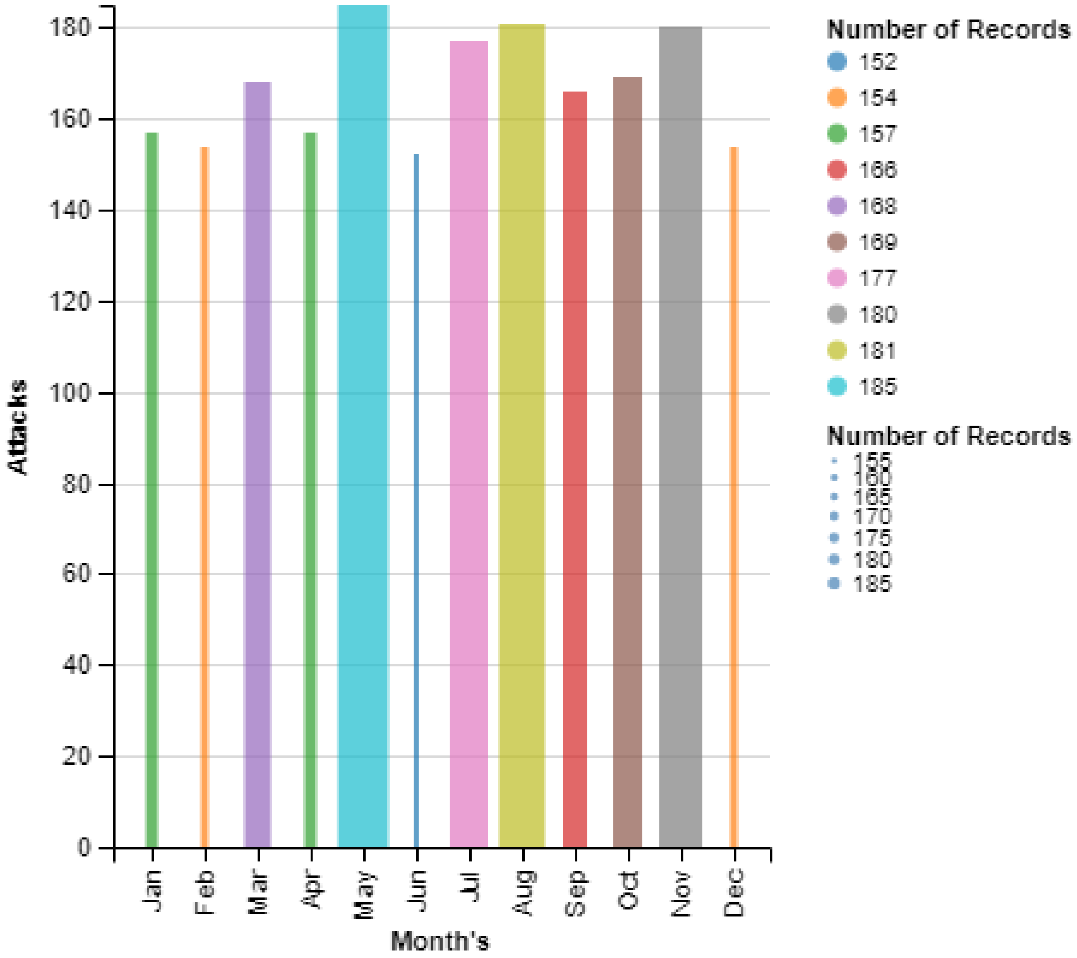
In [16]:

```
alt.Chart(df_sample).mark_line().encode(
    x='year:T',
    y=alt.Y('count(*)',title='Attacks'),
)
# from the graph we can see that 2014 was the highest
# We can check by df_sample.year.value_counts().head()
```



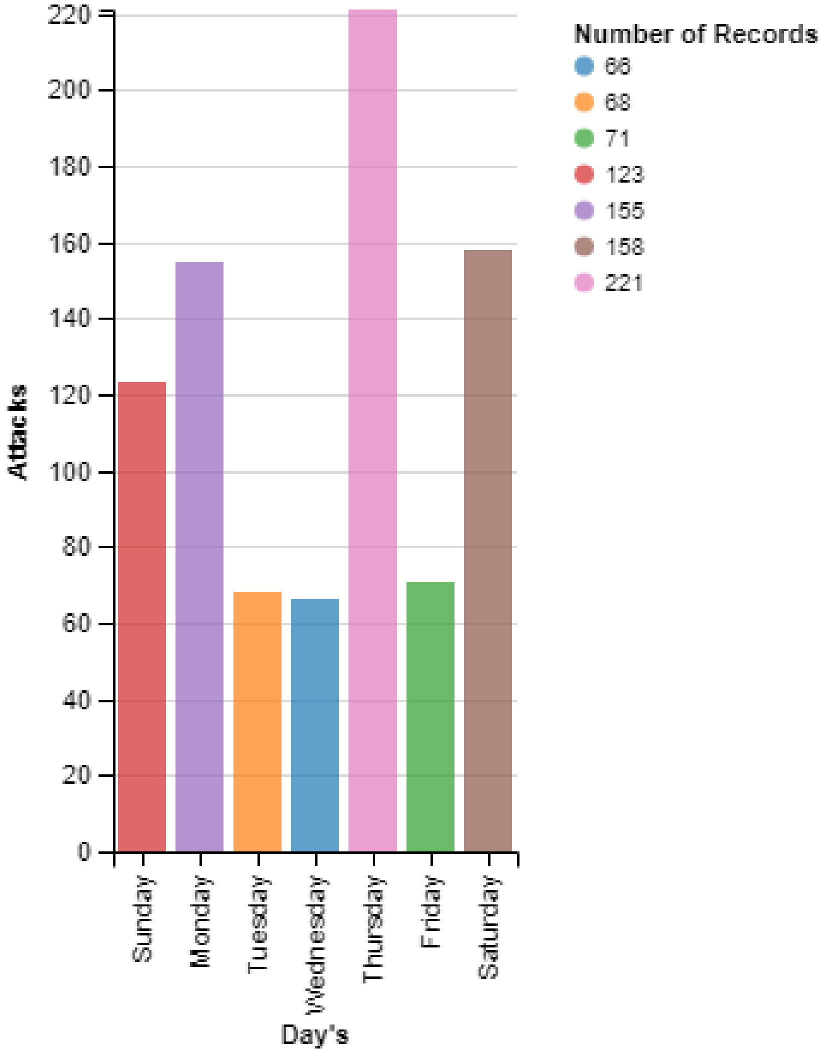
In [143]:

```
alt.Chart(df_sample).mark_bar().encode(
    x=alt.X('month:T', timeUnit='month',title="Month's"),
    y=alt.Y('count(*)',title='Attacks'),
    color='count(*):N',
    size='count(*)'
)
# from the graph we can see that May is the highest
# We can check the top 5 by using df_sample.month.value_counts().head()
```



In [18]:

```
alt.Chart(df_sample).mark_bar().encode(
    x=alt.X('day:T', timeUnit='day',title="Day's"),
    y=alt.Y('count(*)',title='Attacks'),
    color='count(*):N',
)
# We can clearly see that Thursday is the most day of the
# week the strikes occurred in.
```

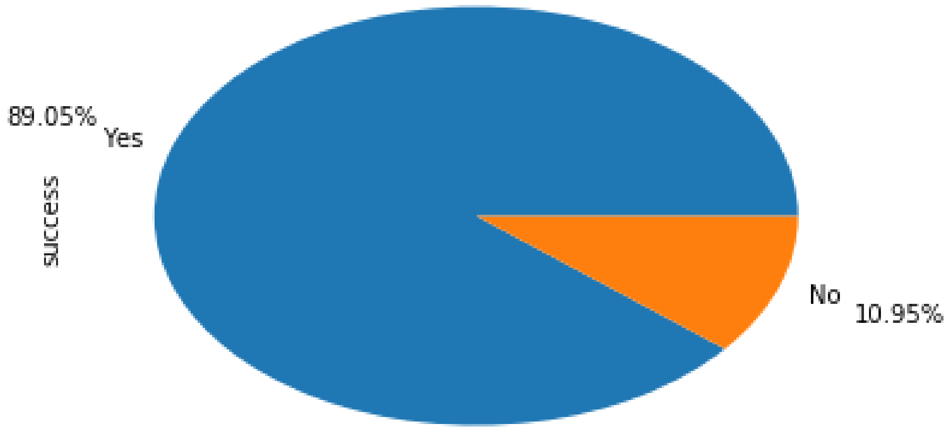


In [19]:

```
df_sample.success.value_counts().plot(kind='pie',autopct='%1.2f%%',pctdista

# 89.05% the operations of the terrorist succeeded
```

Out[19]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99a3adc18>
```

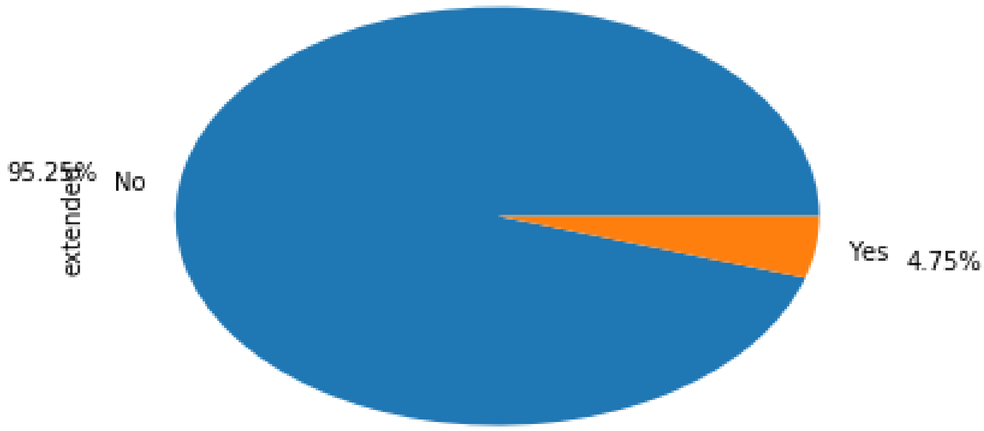


In [20]:

```
df_sample.extended.value_counts().plot(kind='pie',autopct='%1.2f%%',pctdist

# We can see that less than 5% of the strikes in the database
#event's stayed more than 24 hour
```

Out[20]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99a395128>
```

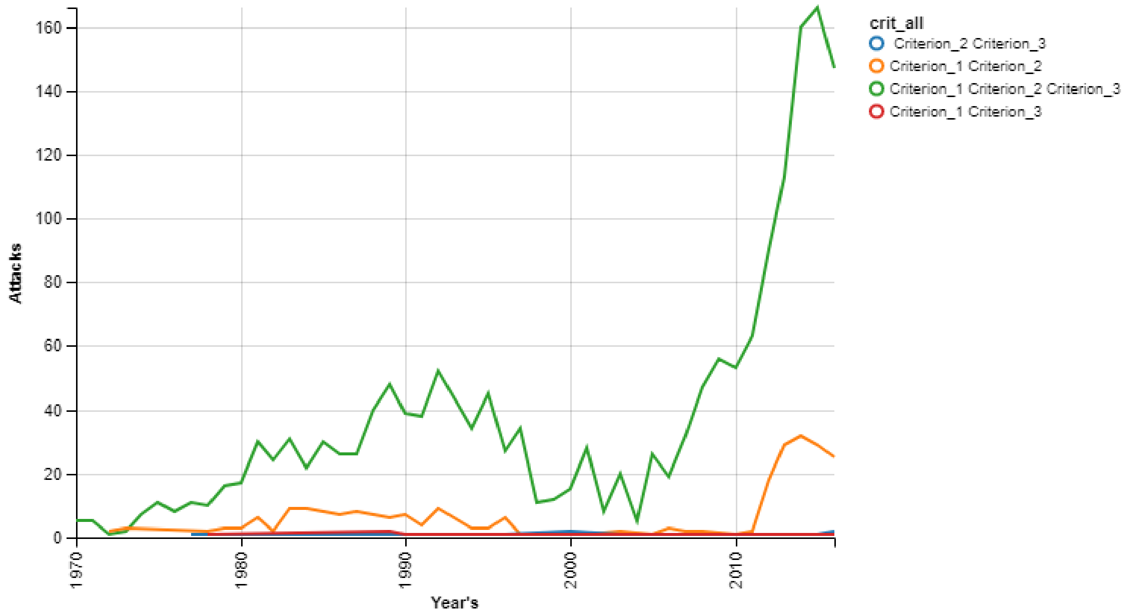


In [22]:

```
df_sample['crit_all']=''

for index, row in df_sample.iterrows():
    x=''
    if row['Criterion_1']=='Yes':
        x='Criterion_1'

    if row['Criterion_2']=='Yes':
        x+=' Criterion_2'

    if row['Criterion_3']=='Yes':
        x+=' Criterion_3'

    df_sample.loc[index,'crit_all']=x

#Create new column out of three to have a better comparison
```

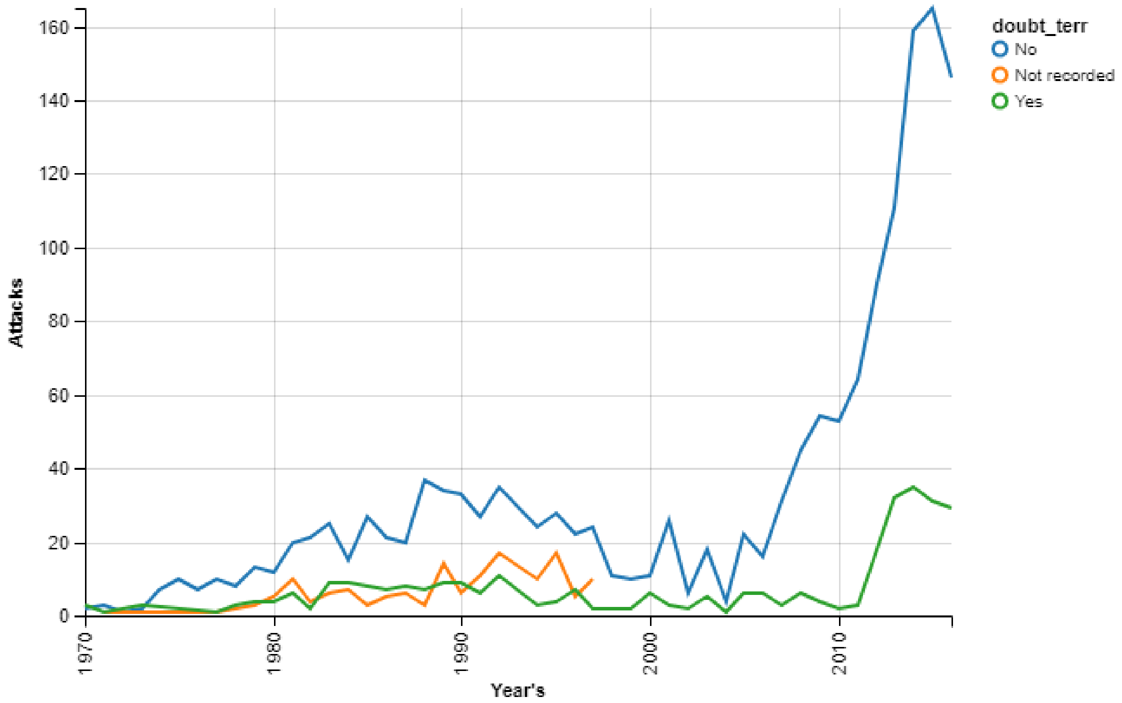
In [144]:

```
alt.Chart(df_sample).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y=alt.Y('count(*)',title='Attacks'),
    color='crit_all',
)
#We can see that most attacks choose all the three:
#Criterion_1 (political, economic, religious, or social goal),
#Criterion_2 (intention to coerce, intimidate, or publicize to larger audie
#Criterion_3 Something outside the international humanitarian law
```



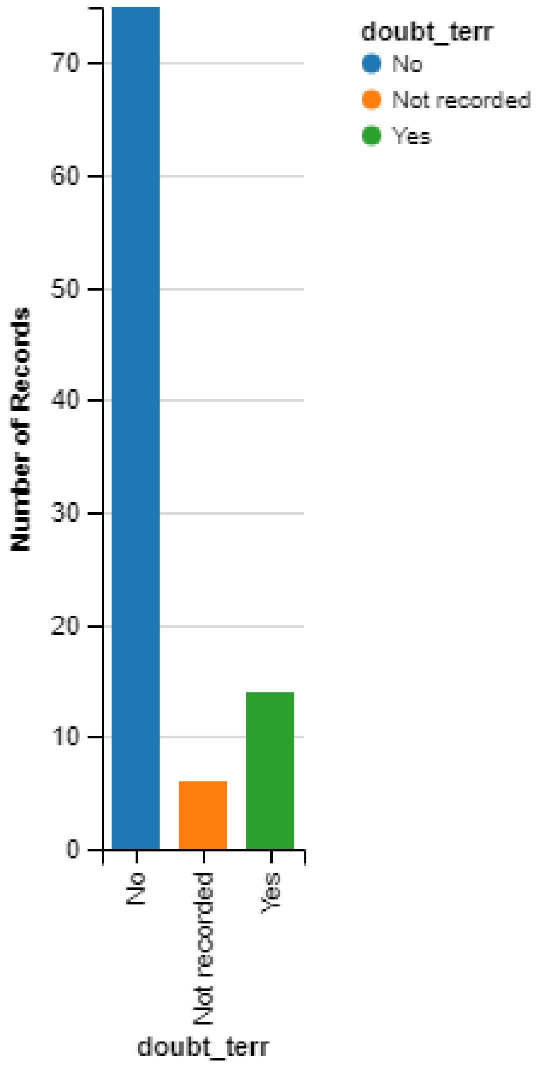
In [24]:

```
alt.Chart(df_sample).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y=alt.Y('count(*)',title='Attacks'),
    color='doubt_terr',
)
# we can see that many event's before 1997 was not recoded if
#it was a terrorist act or not
```



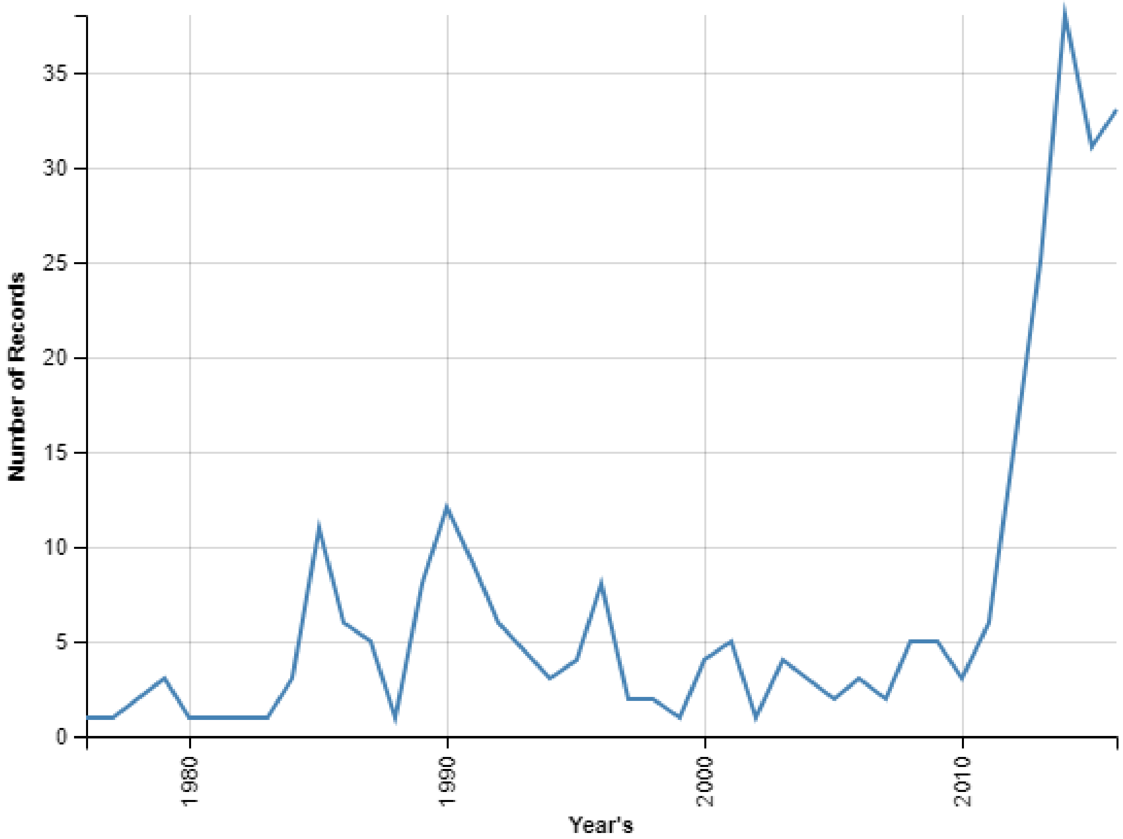
In [25]:

```
alt.Chart(df_sample[df_sample.extended=='Yes']).mark_bar().encode(
    x='doubt_terr',
    y='count(*)',
    color='doubt_terr',
)

# Even after the event's extended more than 24 hour there's
#small amount still doubting if it's a terrorism attack or not
```



In [26]:

```
alt.Chart(df_sample[df_sample.multiple=='Yes']).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y='count(*)',
)

#The event which is part of multiple event's spiked after 2011
```



In [27]:

```
print(df_sample.region[(df_sample.region.notnull())&(df_sample.region!='Unk

top5_attacked_region=df_sample[df_sample.region.isin([
    'Middle East & North Africa',
    'South Asia',
    'South America',
    'Western Europe',
    'Sub-Saharan Africa'])]
```

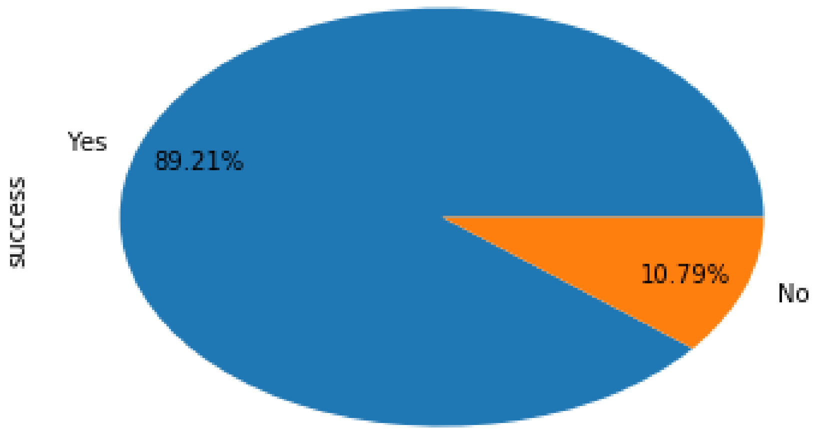
```
South Asia                    525
Middle East & North Africa    521
Sub-Saharan Africa            211
South America                 203
Western Europe                181
Name: region, dtype: int64
```

In [28]:

```
top5_attacked_region.success.value_counts().plot(kind='pie',autopct='%1.2f%
```

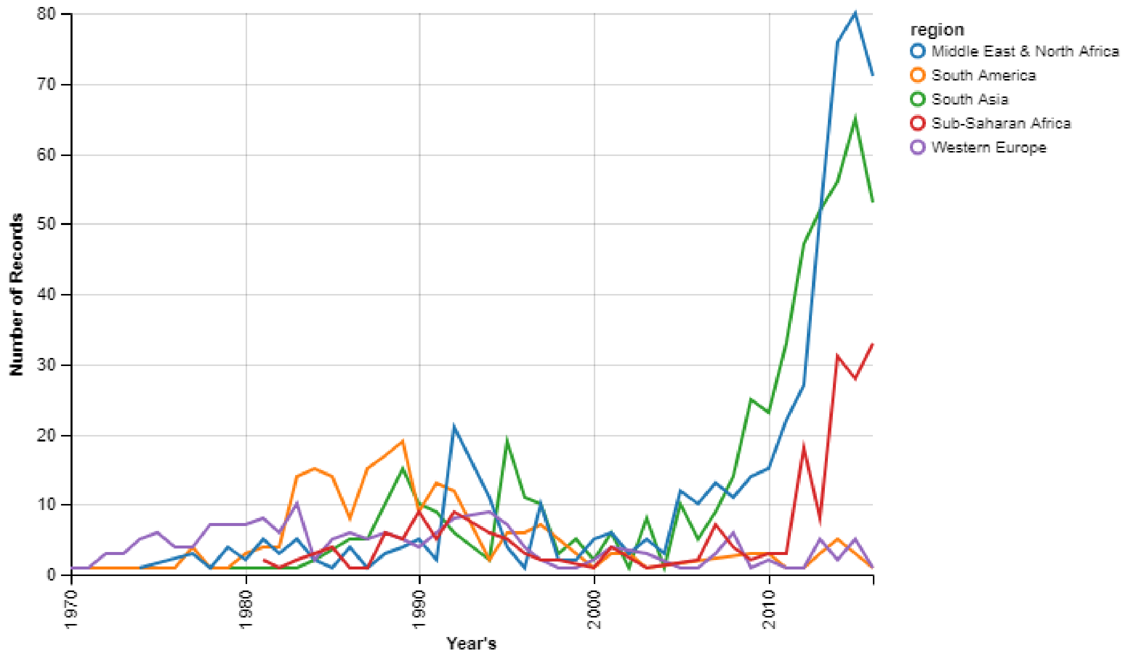
Out[28]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99b547438>
```



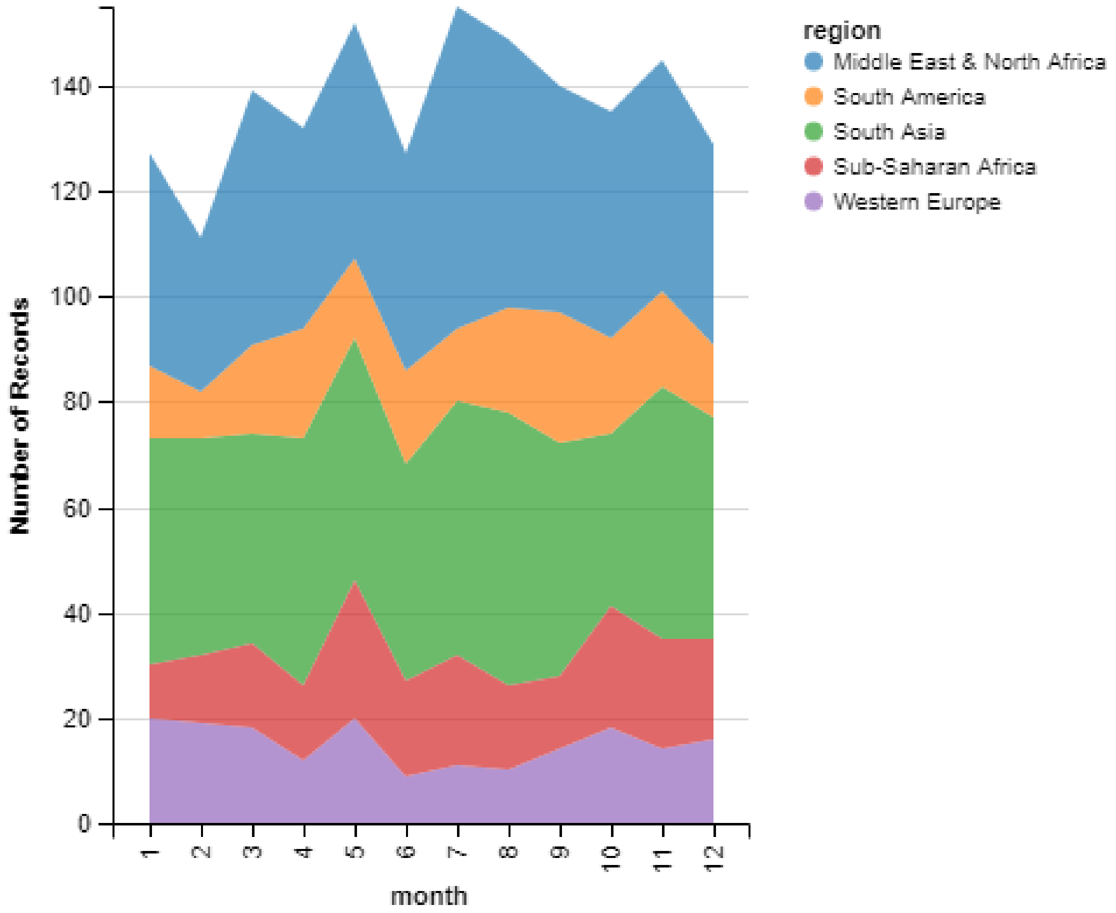
In [29]:

```
alt.Chart(top5_attacked_region).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y='count(*)',
    color='region'
)
# The Middle East and North Africa are the most attacked region
```



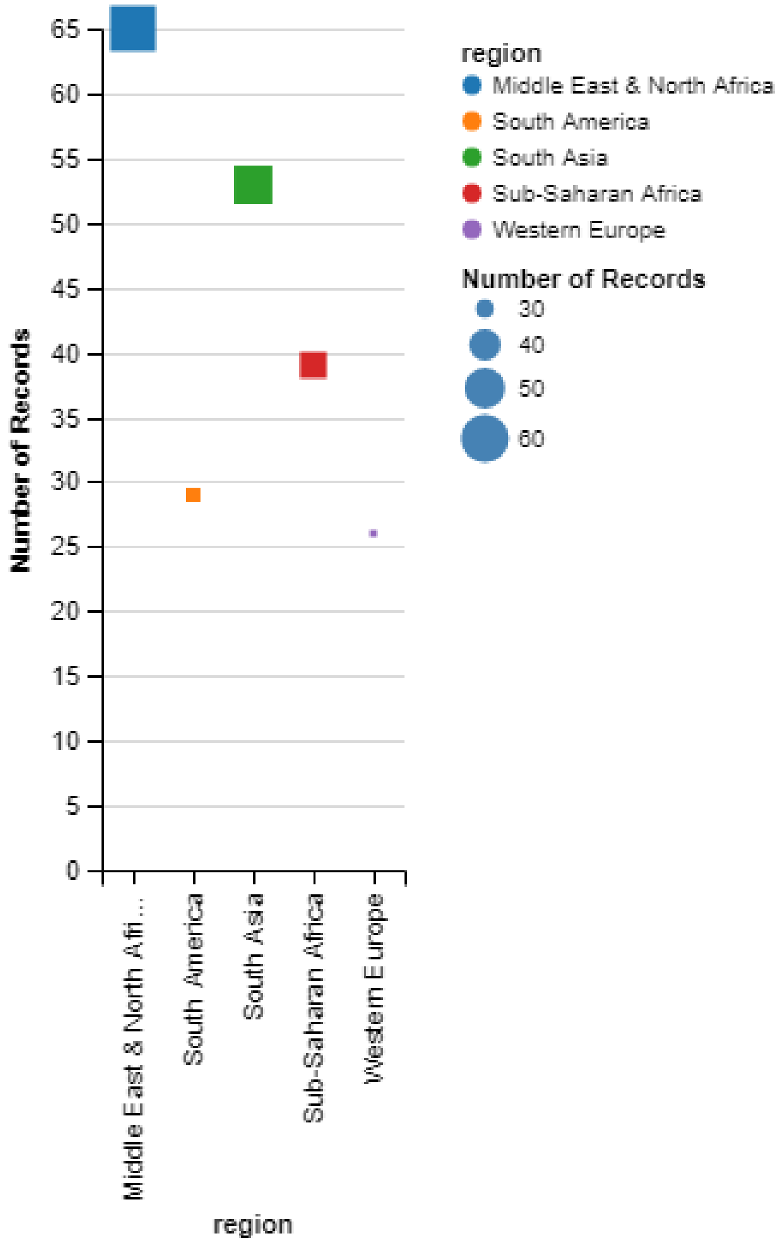
In [30]:

```
alt.Chart(top5_attacked_region).mark_area().encode(
    x='month:N',
    y='count(*)',
    color='region',
    size='count(*)'
)

# We can see the most active month in the Middle east & North Africa is jul
```



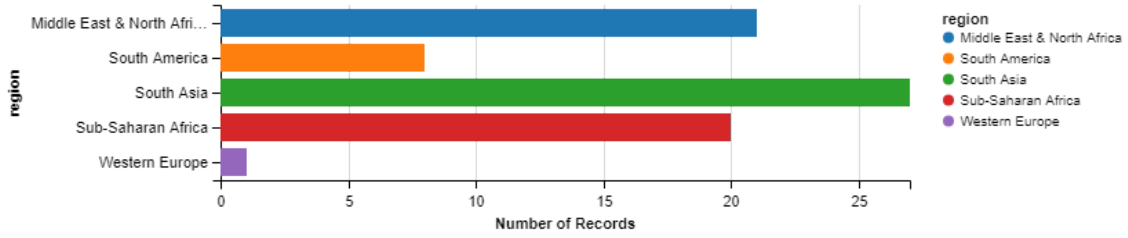
In [31]:

```
alt.Chart(top5_attacked_region[top5_attacked_region.multiple=='Yes']).mark_
    x='region',
    y='count(*)',
    color='region',
    size='count(*)',
)

# Middle East & North Africa got the most attacks
```



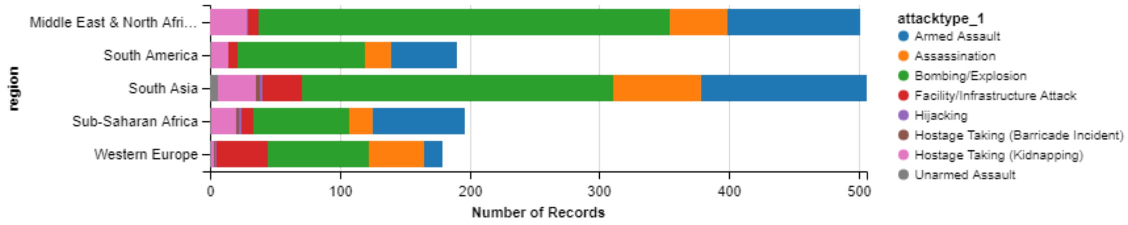
In [32]:

```
alt.Chart(top5_attacked_region[top5_attacked_region.extended=='Yes']).mark_
    x='count(*)',
    y='region',
    color='region',
)
# We can see that South Asia are the most region with events continue for m
```



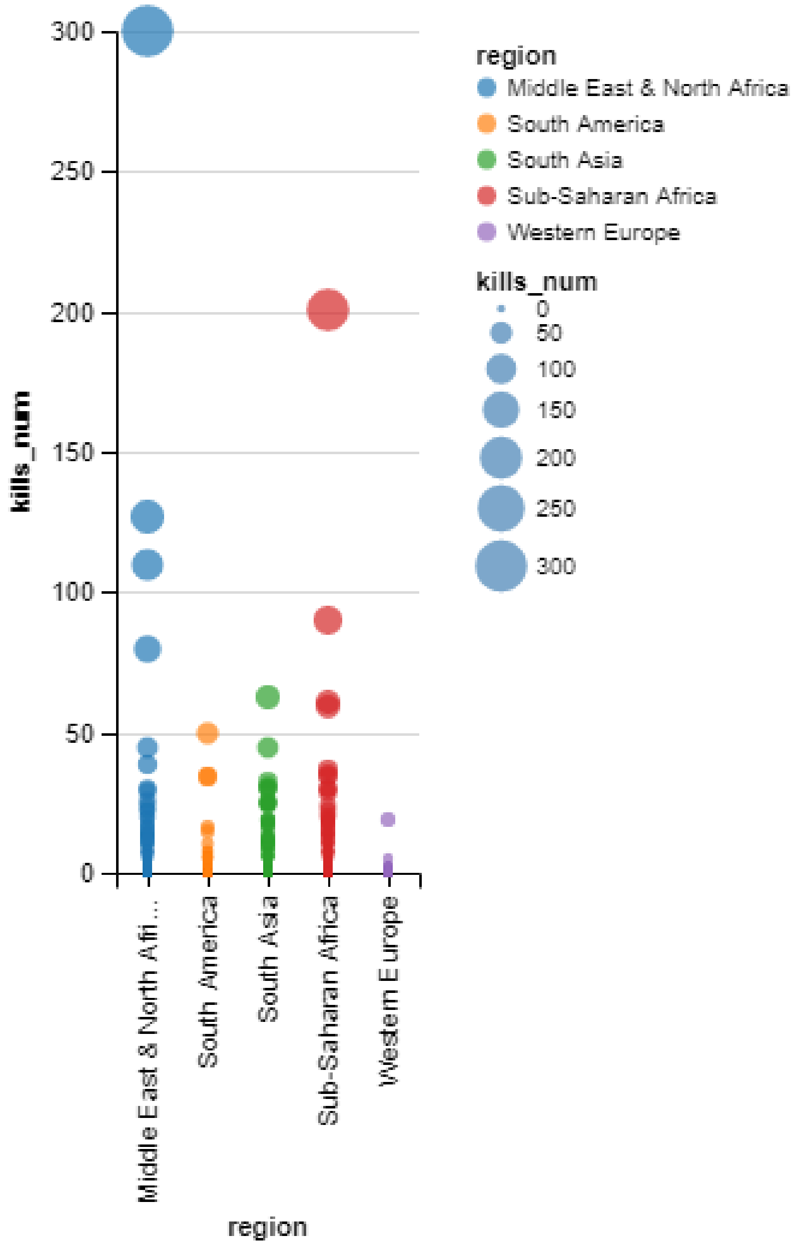
In [33]:

```
alt.Chart(top5_attacked_region[
    (top5_attacked_region.attacktype_1.notnull())&(top5_attacked_region.att
    x='count(*)',
    y='region',
    color='attacktype_1',
)

#The bombing mostly in South Asia and (middle east & north africa)
```



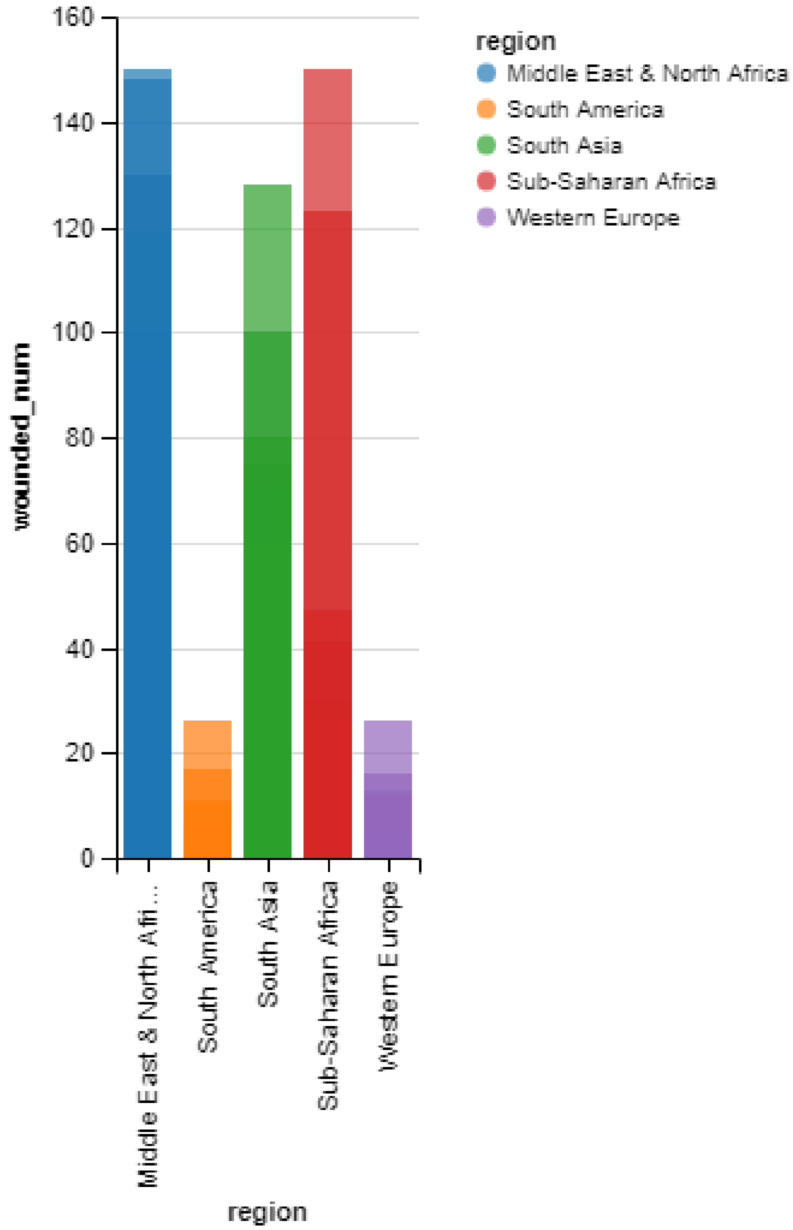
In [34]:

```
alt.Chart(top5_attacked_region[top5_attacked_region.region.notnull()]).mark
    x='region',
    y='kills_num',
    color='region',
    size='kills_num',
)

#The highest region got the most kills numbers: 1st Middle east & north afr
```



In [35]:

```
alt.Chart(top5_attacked_region[(
    top5_attacked_region.region.notnull())&(top5_attacked_region.wounded_nu
    x='region',
    y='wounded_num',
    color='region',
)

# (Middle East & North Africa) equals Sub-Saharan Africa in the wounded num
```



In [36]:

```
print(df_sample.country[(df_sample.country.notnull())&(df_sample.country!='

top5_attacked_countries=df_sample[df_sample.country.isin([
    'Iraq',
    'Pakistan',
    'Afghanistan',
    'India',
    'Colombia'])]
```

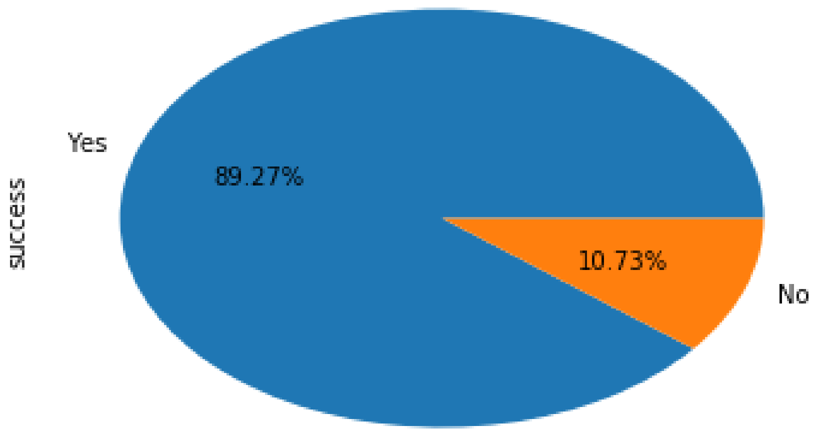
```
Iraq           240
Pakistan       165
India          146
Afghanistan    138
Colombia        94
Name: country, dtype: int64
```

In [38]:

```
top5_attacked_countries.success.value_counts().plot(kind='pie',autopct='%1.
```

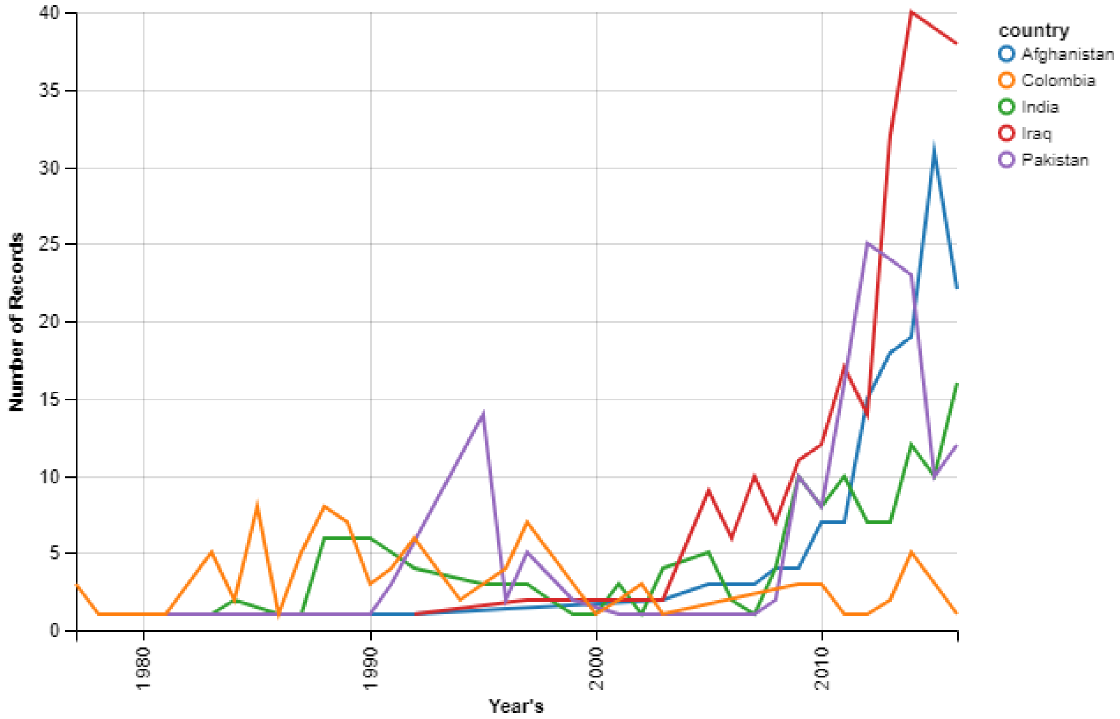
Out[38]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99c3cbfd0>
```



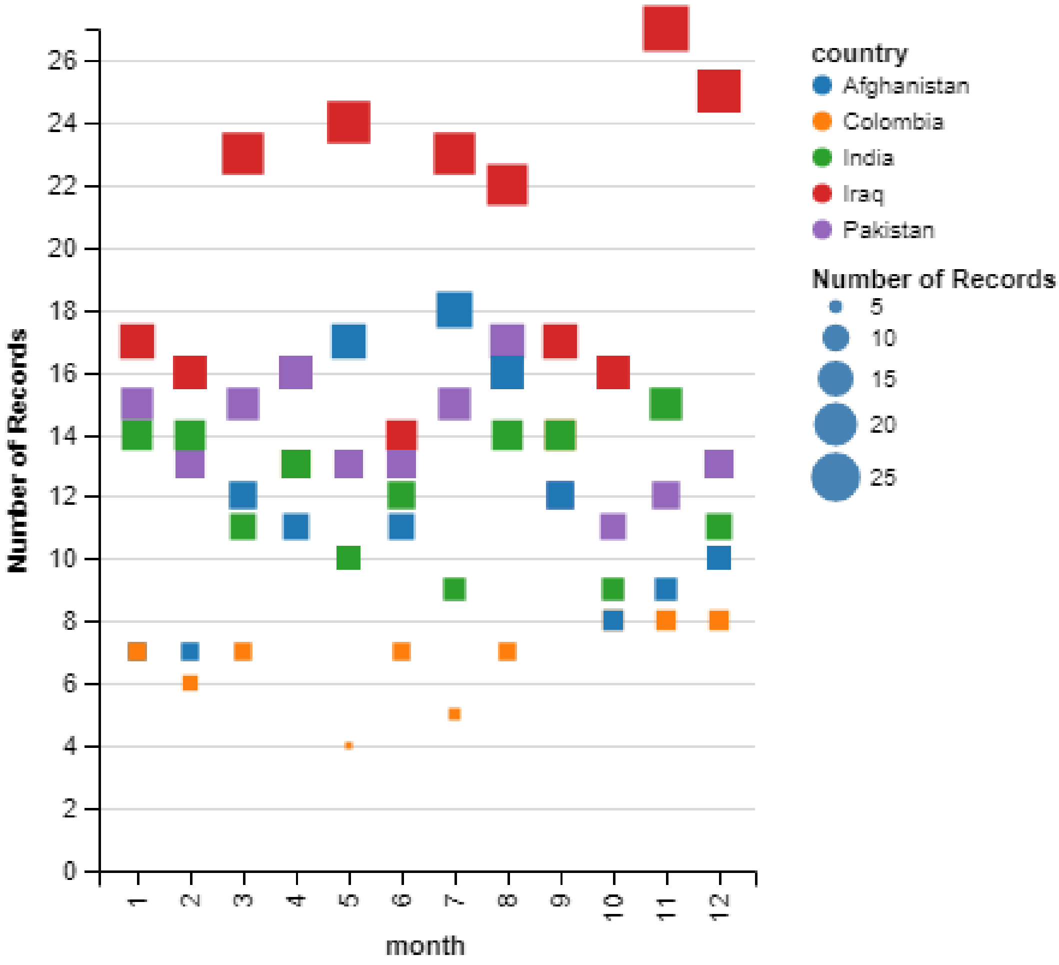
In [39]:

```
alt.Chart(top5_attacked_countries).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y='count(*)',
    color='country'
)
# Iraq attacks increased after saddam death in 2003
```



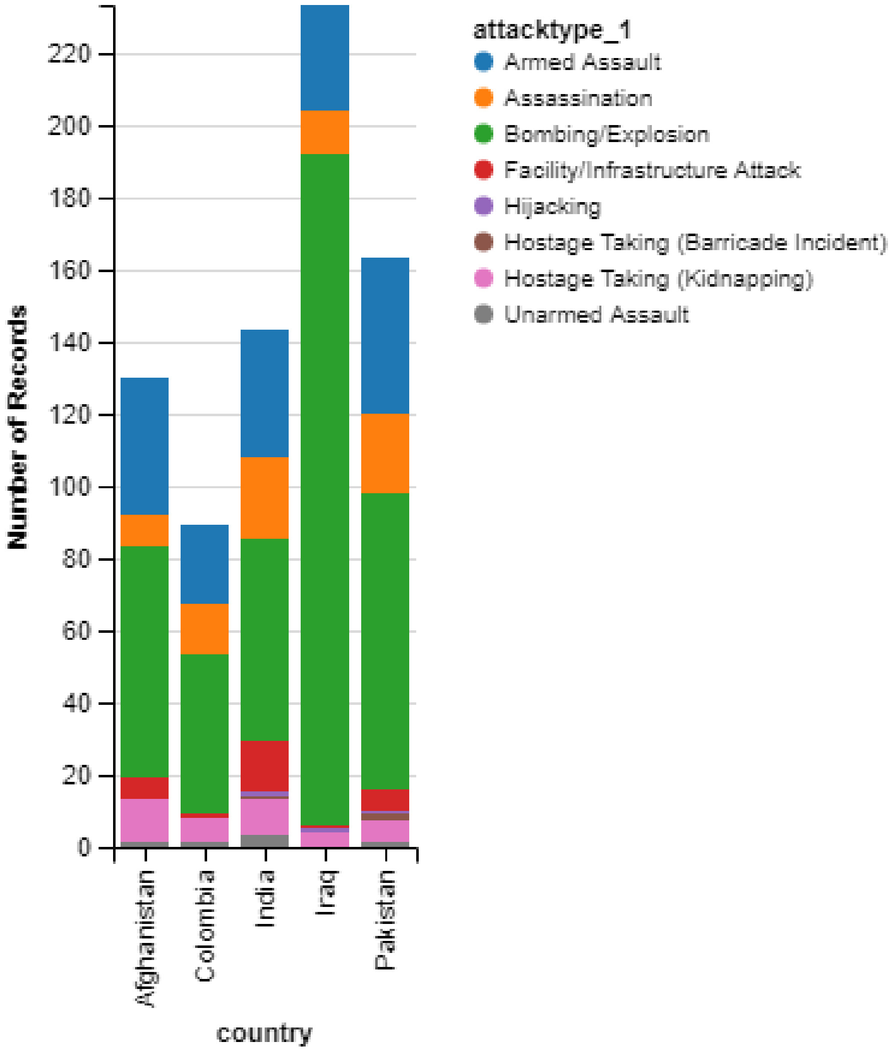
In [40]:

```
alt.Chart(top5_attacked_countries).mark_square().encode(
    x='month:N',
    y='count(*)',
    color='country',
    size='count(*)'
)

# Most attacks happened in november
```



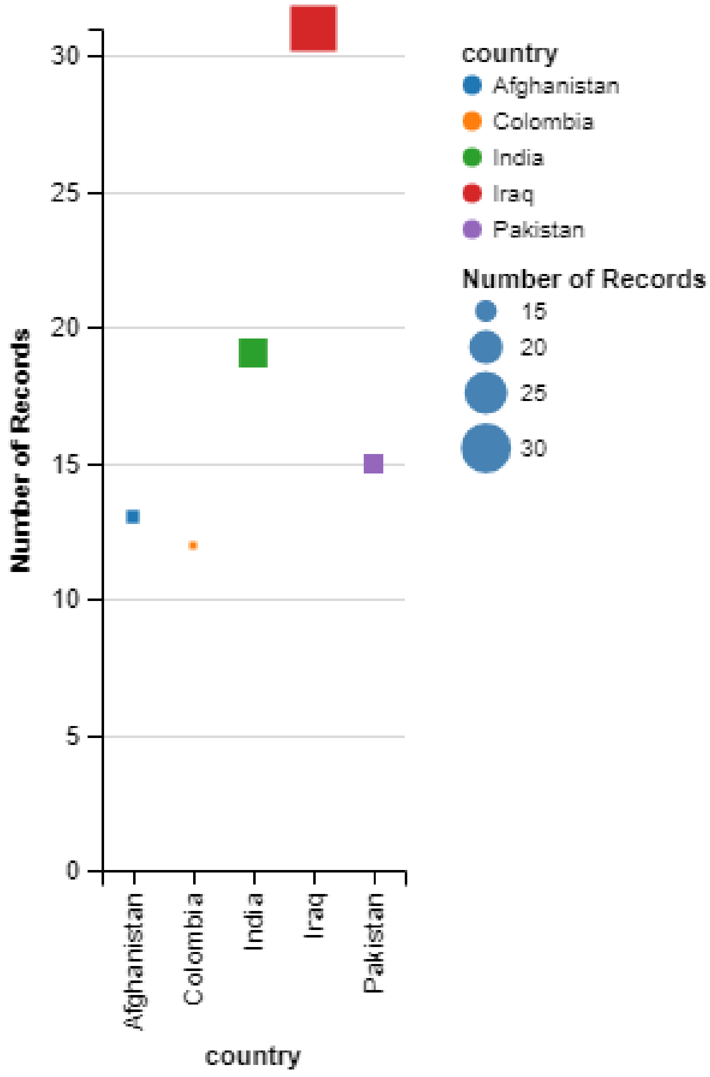
In [37]:

```
alt.Chart(top5_attacked_countries[
    (top5_attacked_countries.attacktype_1.notnull())&(top5_attacked_countri
    x='country',
    y='count(*)',
    color='attacktype_1',
)
# most attack type in iraq: Bombing and the most Armed Assault happened in
```



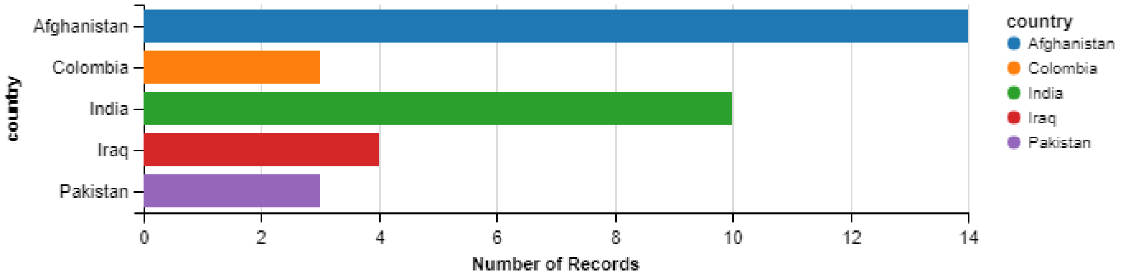
In [41]:

```
alt.Chart(top5_attacked_countries[top5_attacked_countries.multiple=='Yes'])
    x='country',
    y='count(*)',
    color='country',
    size='count(*)',
)

# Iraq got the most event's which connected to multiple events
```



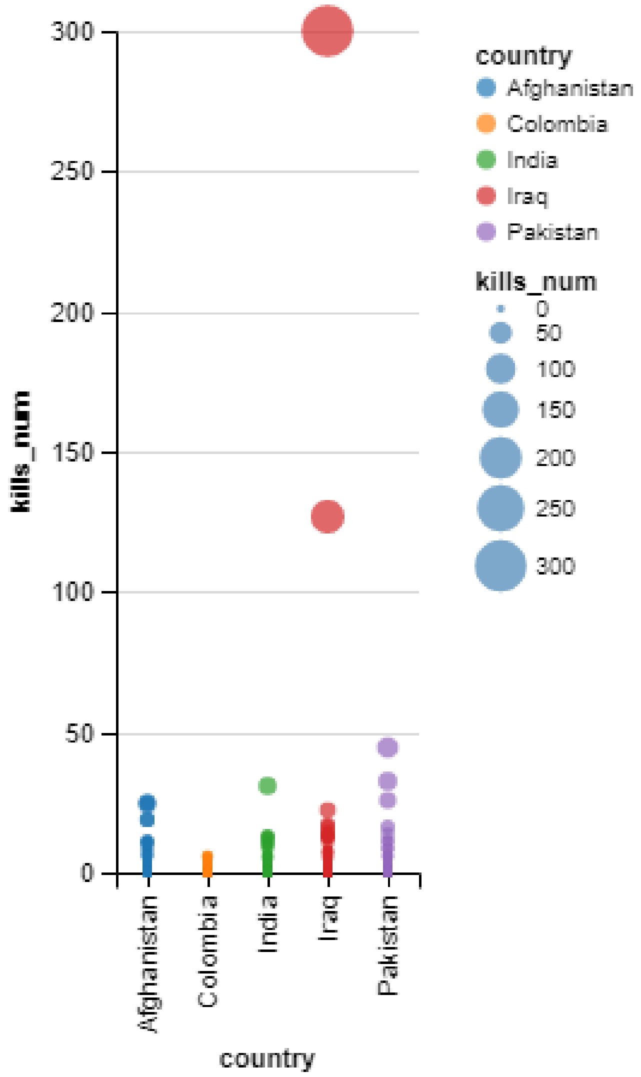
In [42]:

```
alt.Chart(top5_attacked_countries[top5_attacked_countries.extended=='Yes'])
    x='count(*)',
    y='country',
    color='country',
)
# Afghanistan got the most event's extended for more than 24 hour's
```



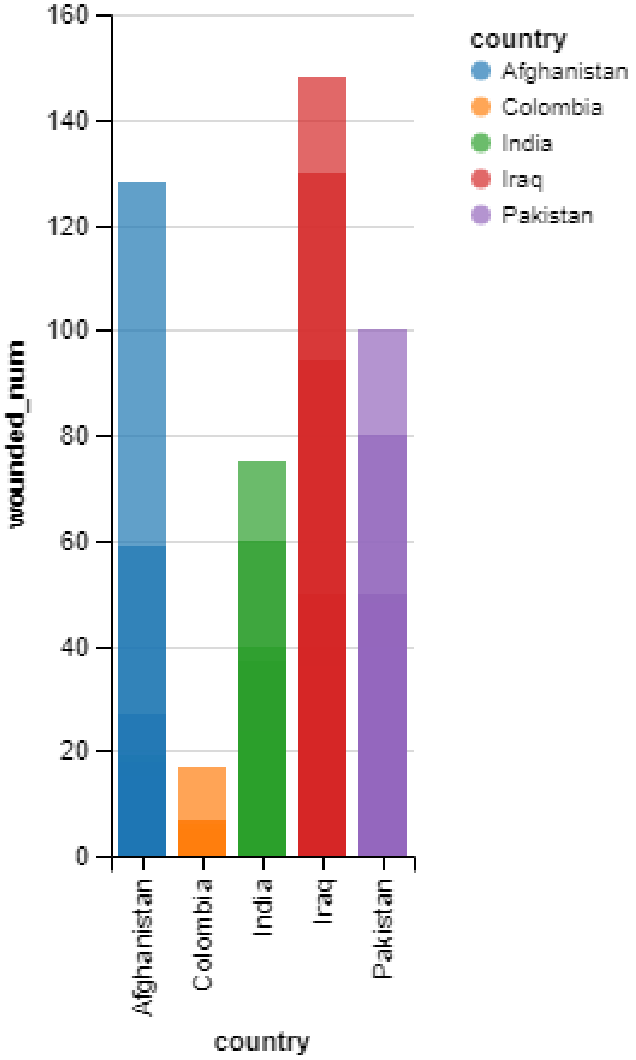
In [43]:

```
alt.Chart(top5_attacked_countries[top5_attacked_countries.kills_num.notnull
    x='country',
    y='kills_num',
    color='country',
    size='kills_num',
)

# Most country got killed people: 1st Iraq
```



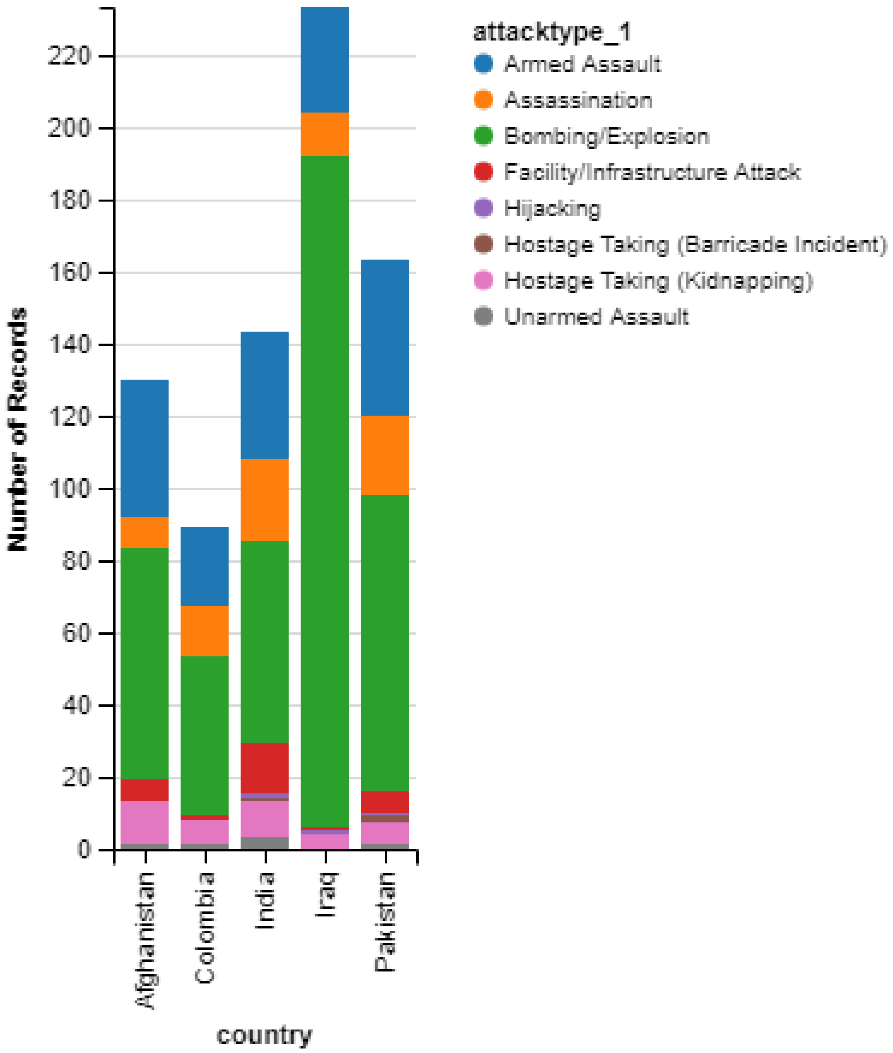
In [44]:

```
alt.Chart(top5_attacked_countries[top5_attacked_countries.wounded_num.notnu
    x='country',
    y='wounded_num',
    color='country',
)

# Iraq got the most wounded people
```



In [45]:

```
alt.Chart(top5_attacked_countries[
    (top5_attacked_countries.attacktype_1.notnull())&(top5_attacked_countri
    x='country',
    y='count(*)',
    color='attacktype_1',
)

#Iraq got explotions the most
```



In [46]:

```
print(df_sample.city[(df_sample.city.notnull())&(df_sample.city!='Unknown')
```

```
Baghdad         70
Karachi         36
Lima            23
San Salvador    22
Belfast         19
Name: city, dtype: int64
```

In [47]:

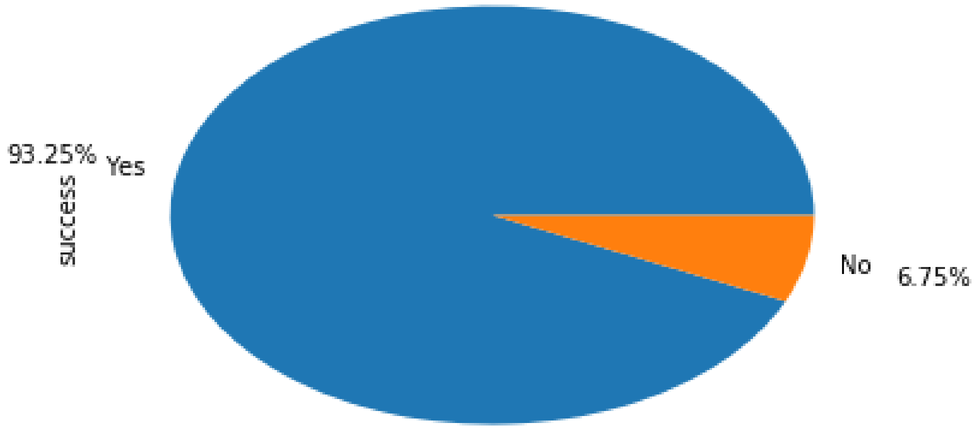
```
top5_attacked_cities=df_sample[df_sample.city.isin([
    'Baghdad',
    'Belfast',
    'Karachi',
    'Lima',
    'Mosul'])]
```

In [48]:

```
top5_attacked_cities.success.value_counts().plot(kind='pie',autopct='%1.2f%
```

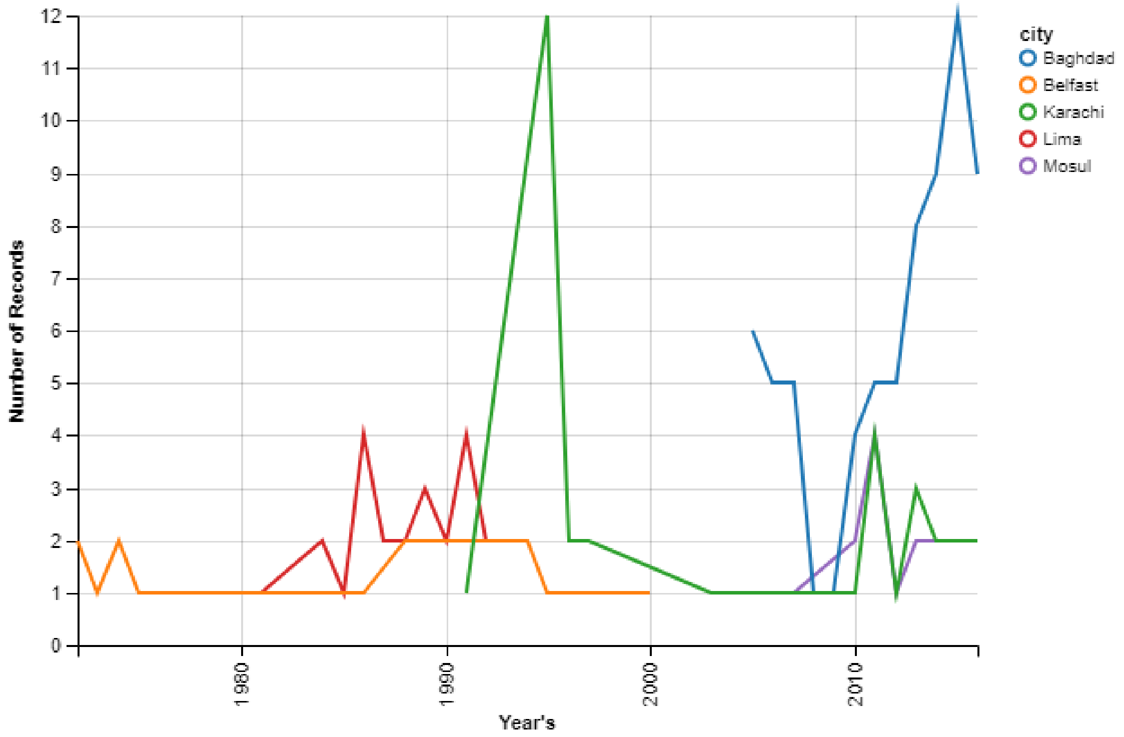
Out[48]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99c3edc88>
```



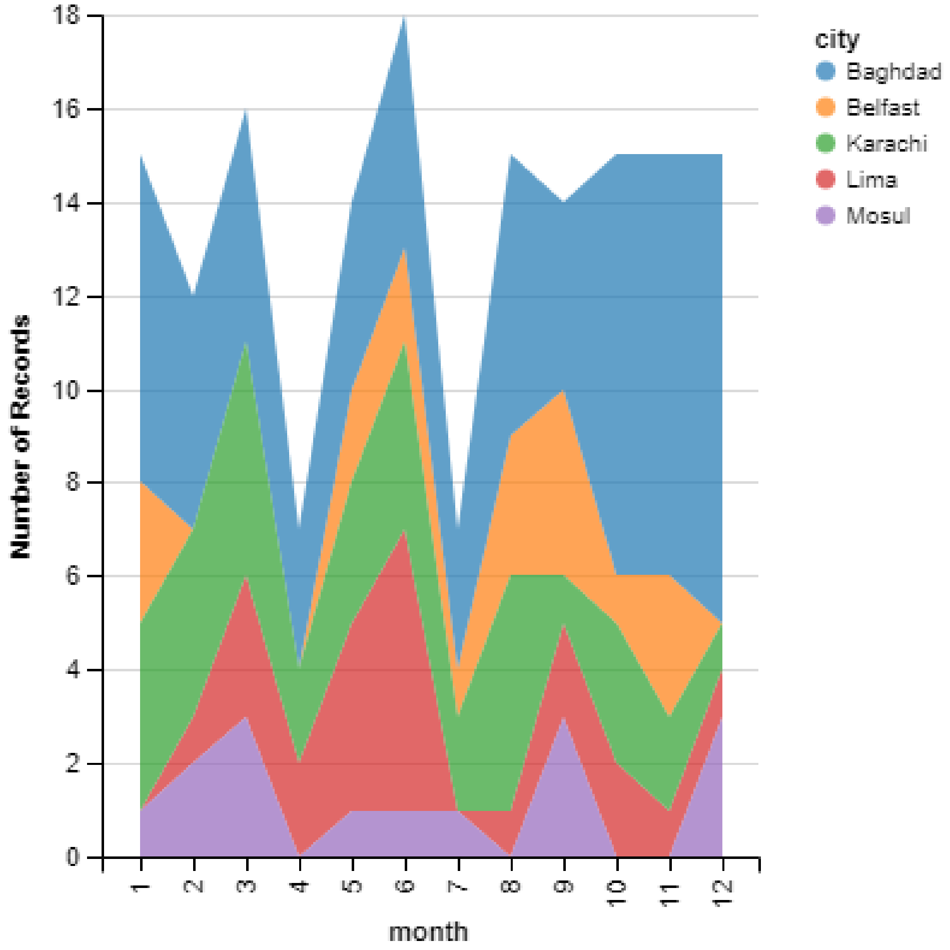
In [49]:

```
alt.Chart(top5_attacked_cities).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y='count(*)',
    color='city'
)
# Baghdad attacks spiked after 1991 and Lima city attacks stopped in 1992
```



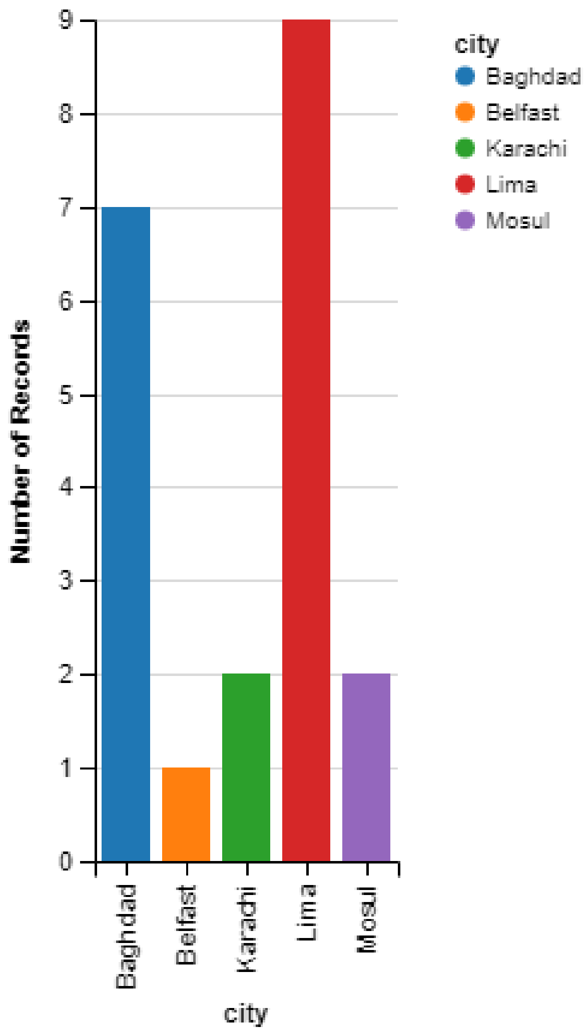
In [50]:

```
alt.Chart(top5_attacked_cities).mark_area().encode(
    x='month:N',
    y='count(*)',
    color='city',
)
# Most of the attacks come in June most of the time
# except for Mosul
```



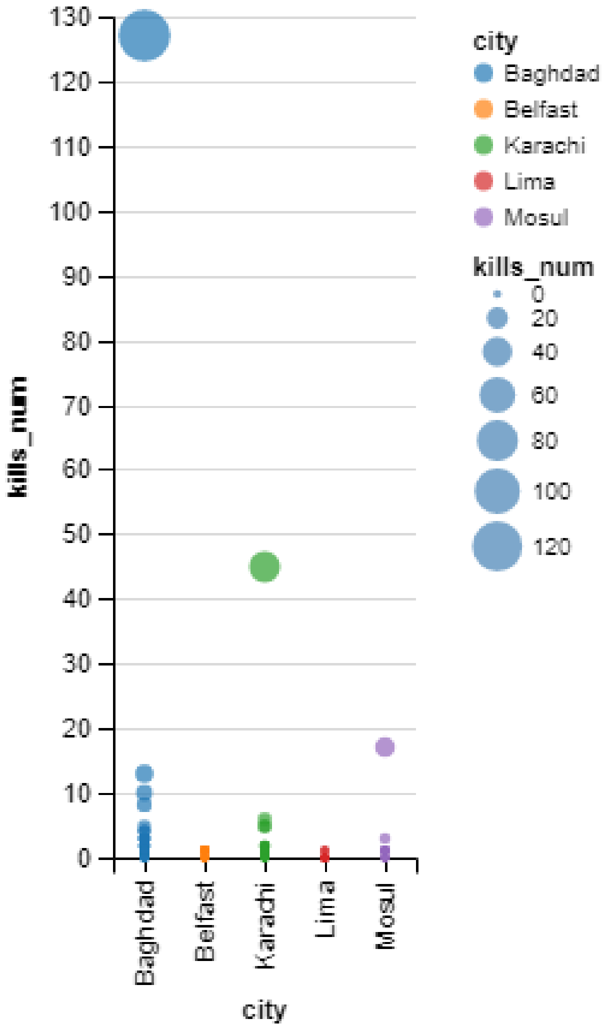
In [115]:

```
alt.Chart(top5_attacked_cities[top5_attacked_cities.multiple=='Yes']).mark_
    x='city',
    y='count(*)',
    color='city',
)

# Lima and Baghdad got the most multiple event's
```



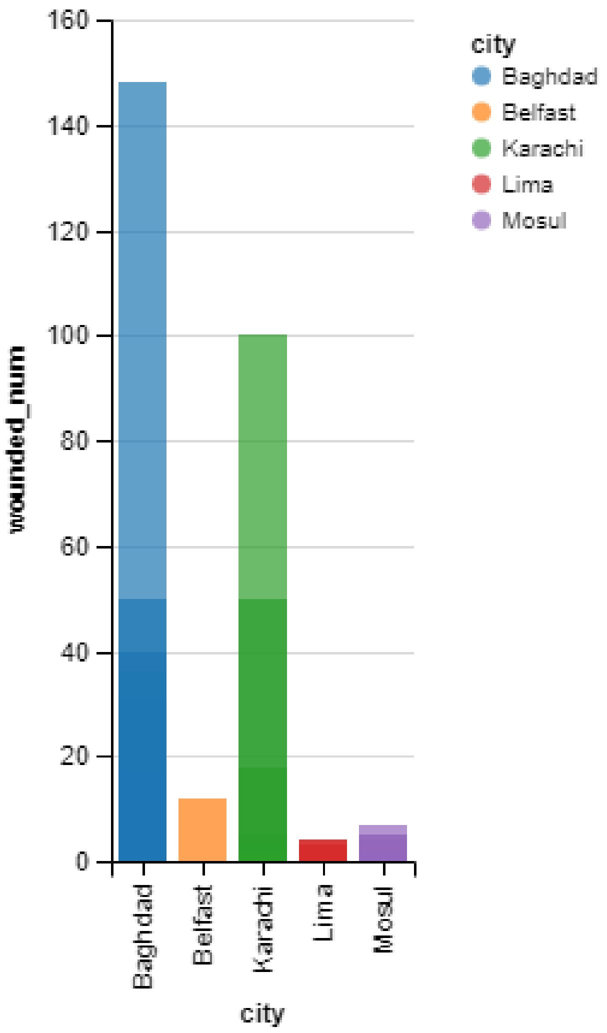
In [53]:

```
alt.Chart(top5_attacked_cities[top5_attacked_cities.kills_num.notnull()]).m
    x='city',
    y='kills_num',
    color='city',
    size='kills_num',
)

#Baghdad got the most killing
```



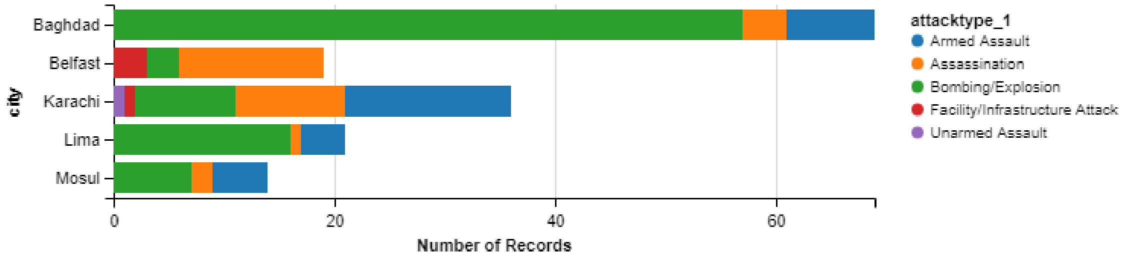
In [54]:

```
alt.Chart(top5_attacked_cities[top5_attacked_cities.wounded_num.notnull()])
    x='city',
    y='wounded_num',
    color='city',
)

#Baghdad got the most wounded people
```



In [55]:

```
alt.Chart(top5_attacked_cities[
    (top5_attacked_cities.attacktype_1.notnull())&(top5_attacked_cities.att
    x='count(*)',
    y='city',
    color='attacktype_1',
)
# Baghdad got bombed the most then Karachi got Armed assault the most
```



In [56]:

```
print(df_sample.nationality[
    (df_sample.nationality.notnull())&(df_sample.nationality!='Unknown')].v

#Here we can see the most targeted nationality
```

```
Iraq           235
Pakistan       155
India          150
Afghanistan    112
Colombia        84
Name: nationality, dtype: int64
```

In [57]:

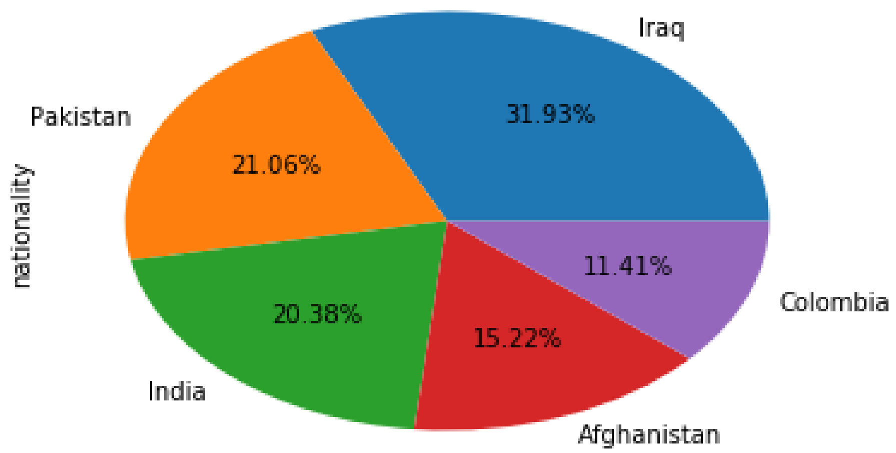
```
top5_target_nationality=df_sample[df_sample.nationality.isin([
    'Iraq',
    'Pakistan',
    'India',
    'Afghanistan',
    'Colombia'])]
```

In [121]:

```
top5_target_nationality.nationality.value_counts().plot(kind='pie',autopct=

# The percentage of the attacks on those nationality
```

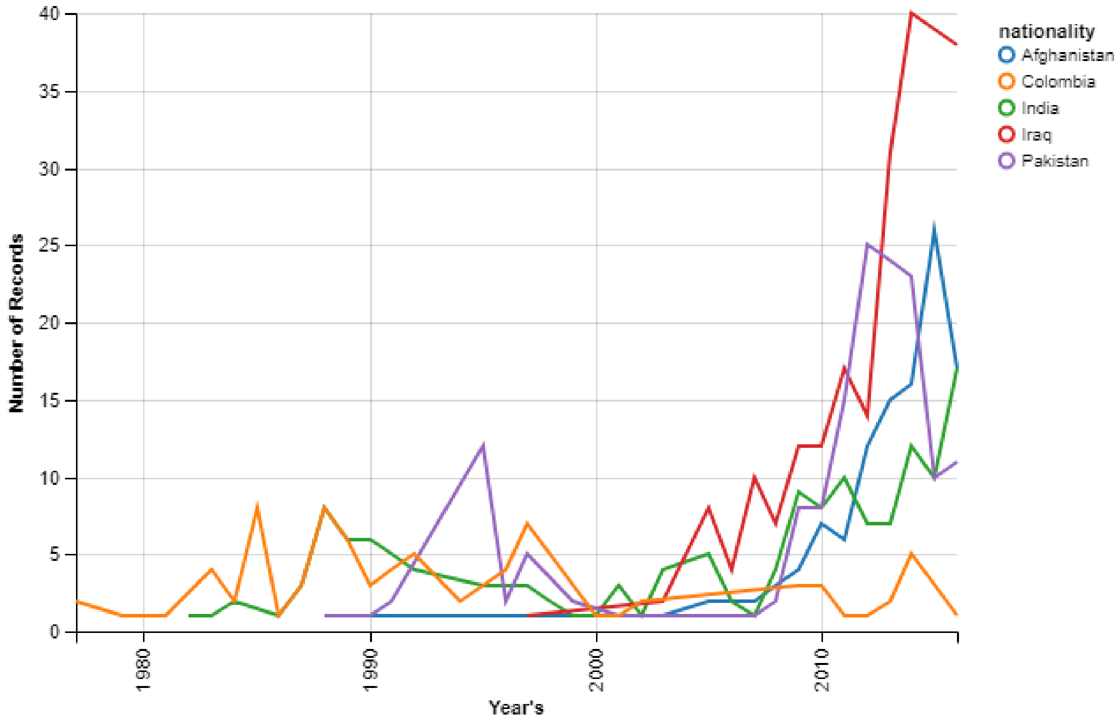
Out[121]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99c4e6780>
```



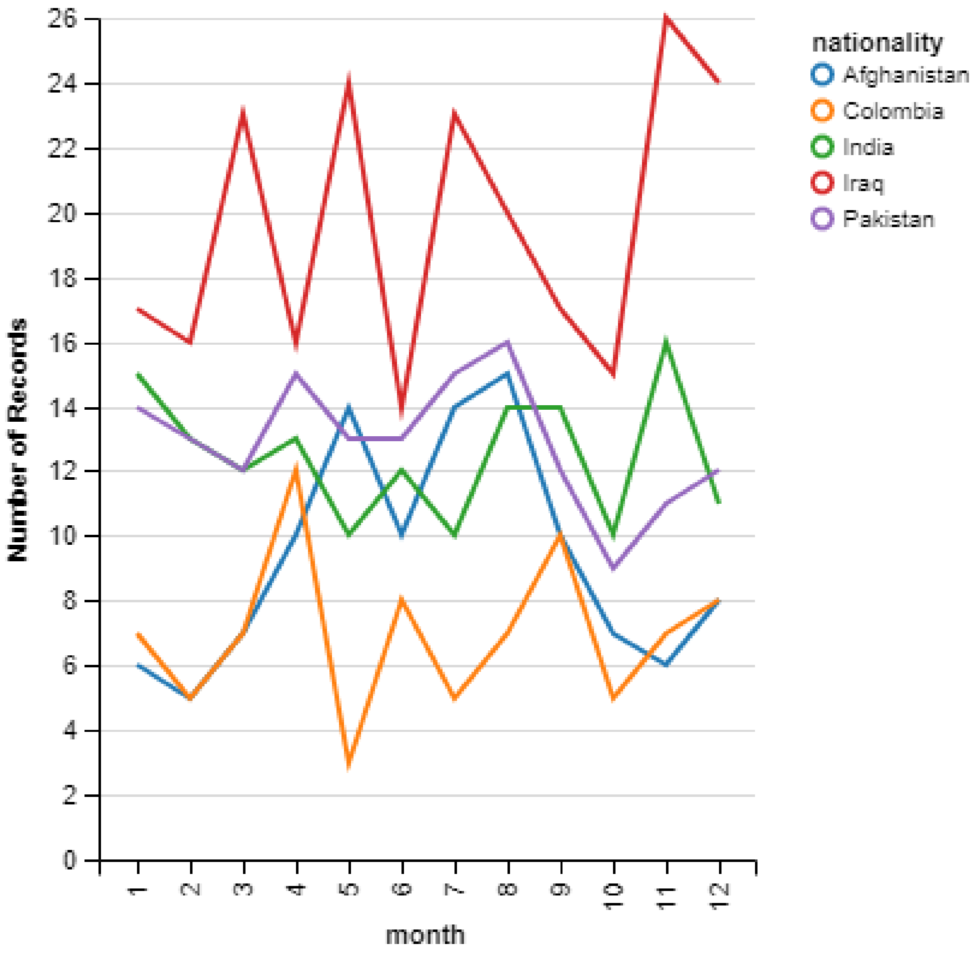
In [59]:

```
alt.Chart(top5_target_nationality).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y='count(*)',
    color='nationality'
)
```



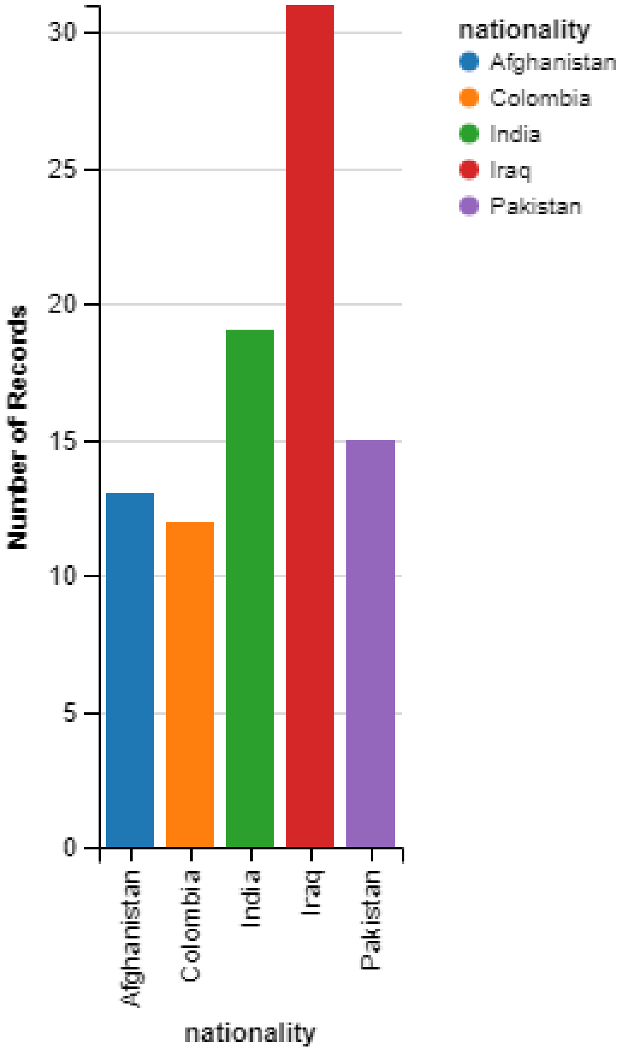
In [60]:

```
alt.Chart(top5_target_nationality).mark_line().encode(
    x='month:N',
    y='count(*)',
    color='nationality',
    size='count(*)'
)
```



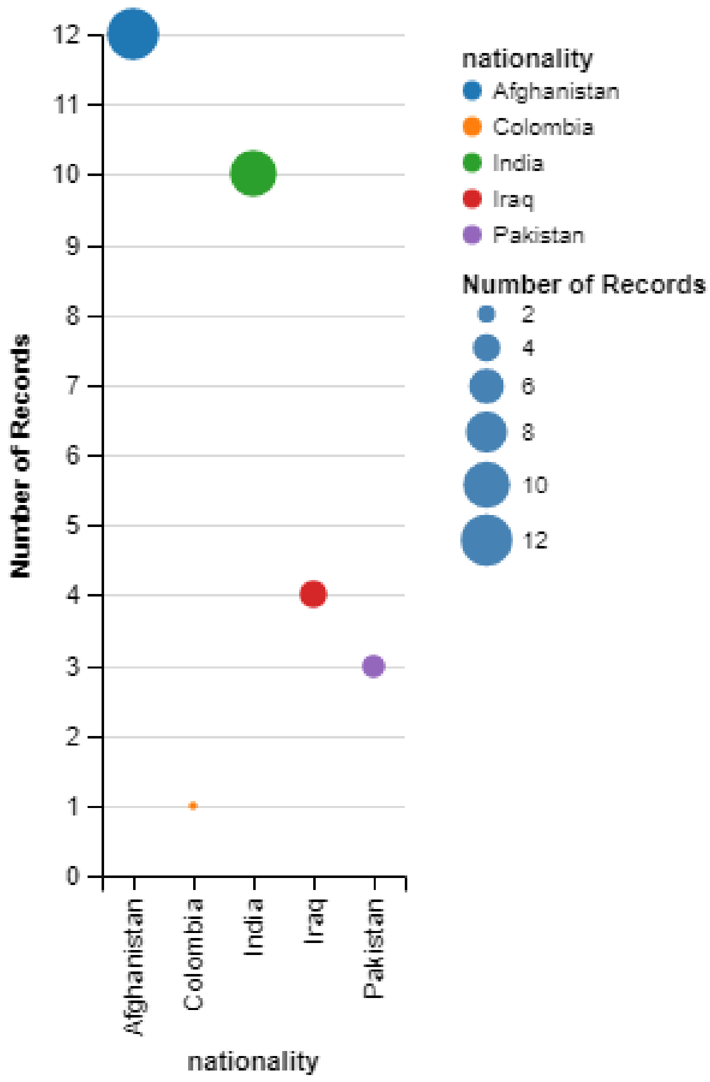
In [61]:

```
alt.Chart(top5_target_nationality[top5_target_nationality.multiple=='Yes'])
    x='nationality',
    y='count(*)',
    color='nationality',
)

# The highest nationalities who got multiple event's are Iraqis and India
```



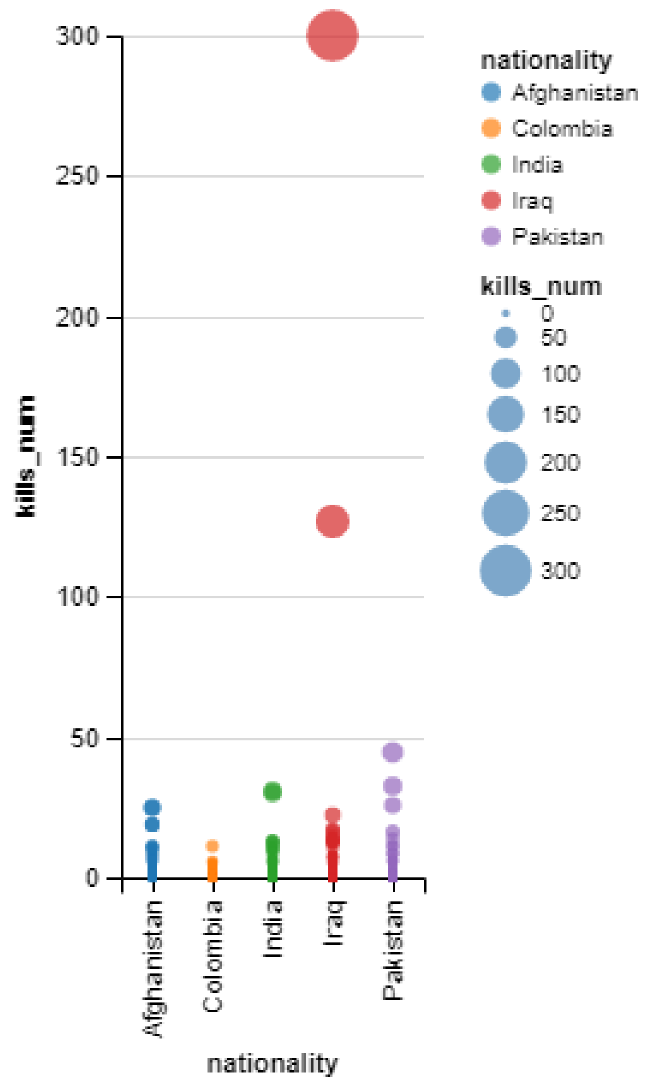
In [62]:

```
alt.Chart(top5_target_nationality[top5_target_nationality.extended=='Yes'])
    x='nationality',
    y='count(*)',
    color='nationality',
    size='count(*)'
)
# The highest nationalities who got Extended event's are Afghanistan and In
```



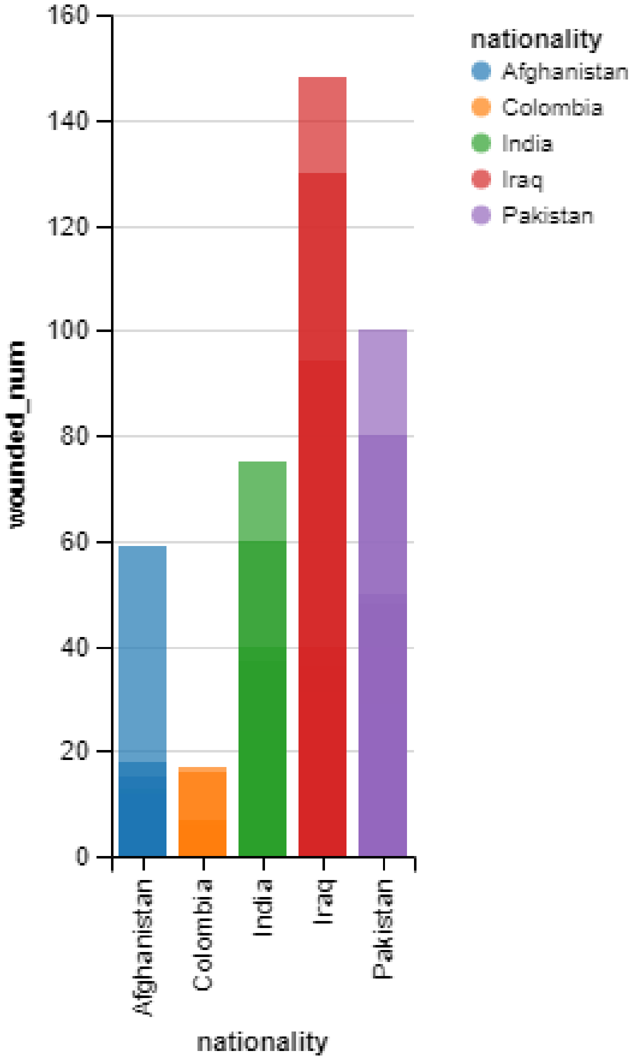
In [63]:

```
alt.Chart(top5_target_nationality[top5_target_nationality.kills_num.notnull
    x='nationality',
    y='kills_num',
    color='nationality',
    size='kills_num',
)

# Iraqi people got the highest number of kills
```



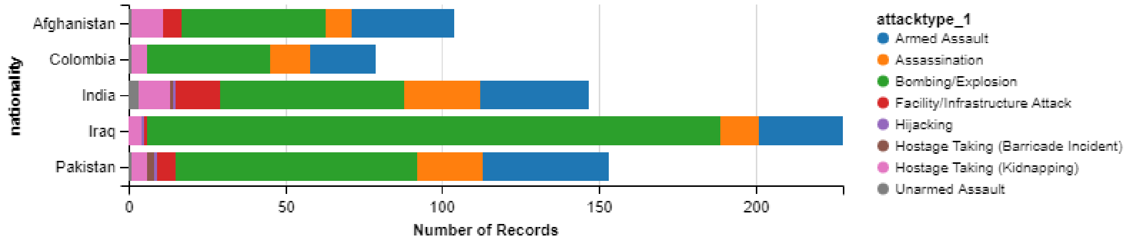
In [64]:

```
alt.Chart(top5_target_nationality[top5_target_nationality.wounded_num.notnu
    x='nationality',
    y='wounded_num',
    color='nationality',
)

# The Indians got killed the most
```



In [65]:

```
alt.Chart(top5_target_nationality[top5_target_nationality.attacktype_1!='Un
    x='count(*)',
    y='nationality',
    color='attacktype_1',
)

# We can see Iraqis got Bombed most of the time
```



In [66]:

```
print(df_sample.attacktype_1[(df_sample.attacktype_1.notnull())&(df_sample.

top5_attacktype=df_sample[df_sample.attacktype_1.isin([
    'Bombing/Explosion',
    'Armed Assault',
    'Assassination',
    'Hostage Taking (Kidnapping)',
    'Facility/Infrastructure Attack'])]
```

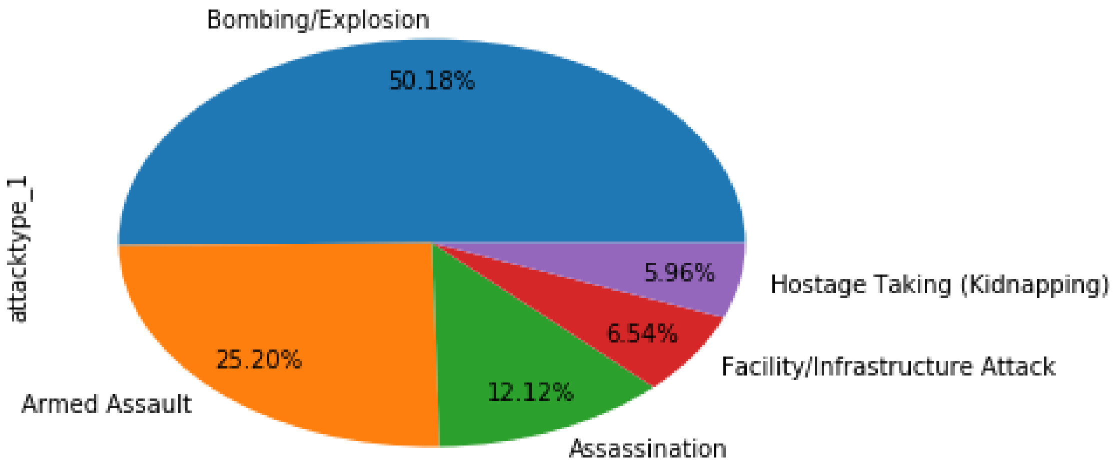
```
Bombing/Explosion                 952
Armed Assault                     478
Assassination                     230
Facility/Infrastructure Attack    124
Hostage Taking (Kidnapping)       113
Name: attacktype_1, dtype: int64
```

In [67]:

```
top5_attacktype.attacktype_1.value_counts().plot(kind='pie',autopct='%1.2f%

#The most attack type used from all the event's are Bombing
```

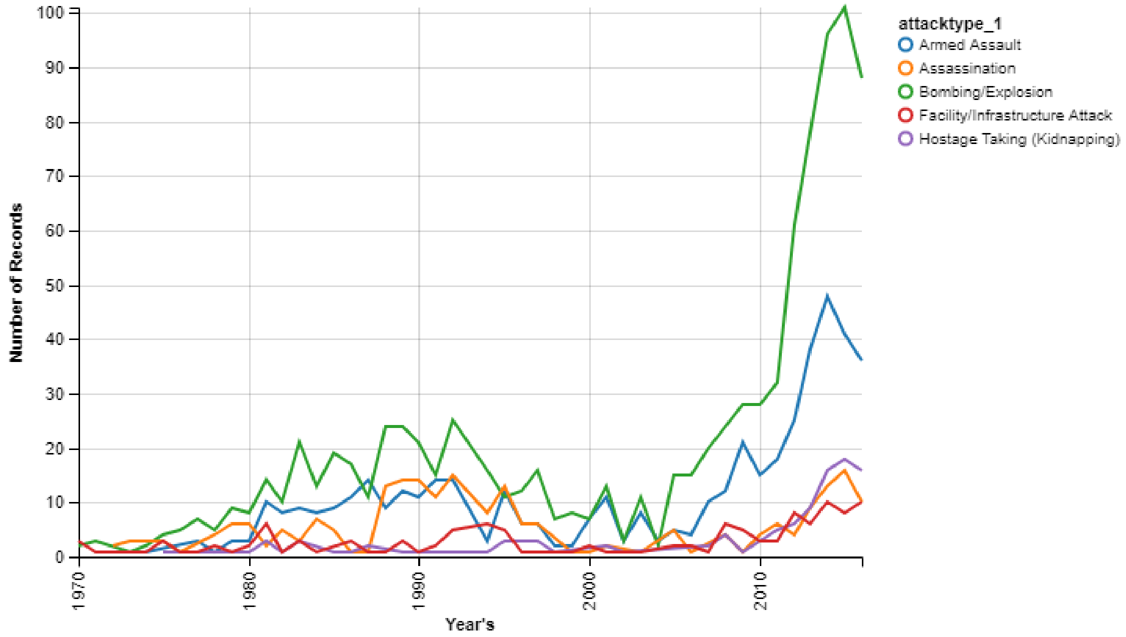
Out[67]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99c4352b0>
```



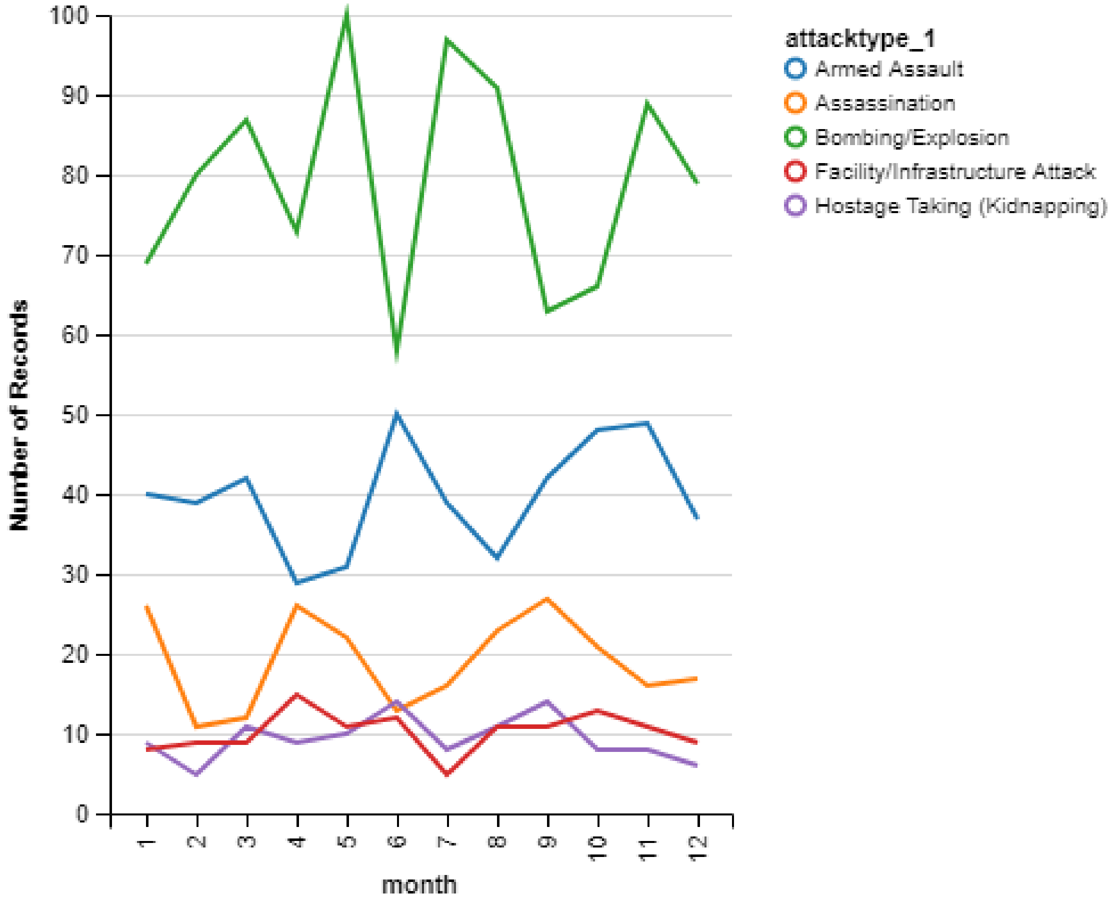
In [68]:

```
alt.Chart(top5_attacktype).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y='count(*)',
    color='attacktype_1'
)
# The 1st one Bombing/Explosion and the 2nd Armed Assault
```



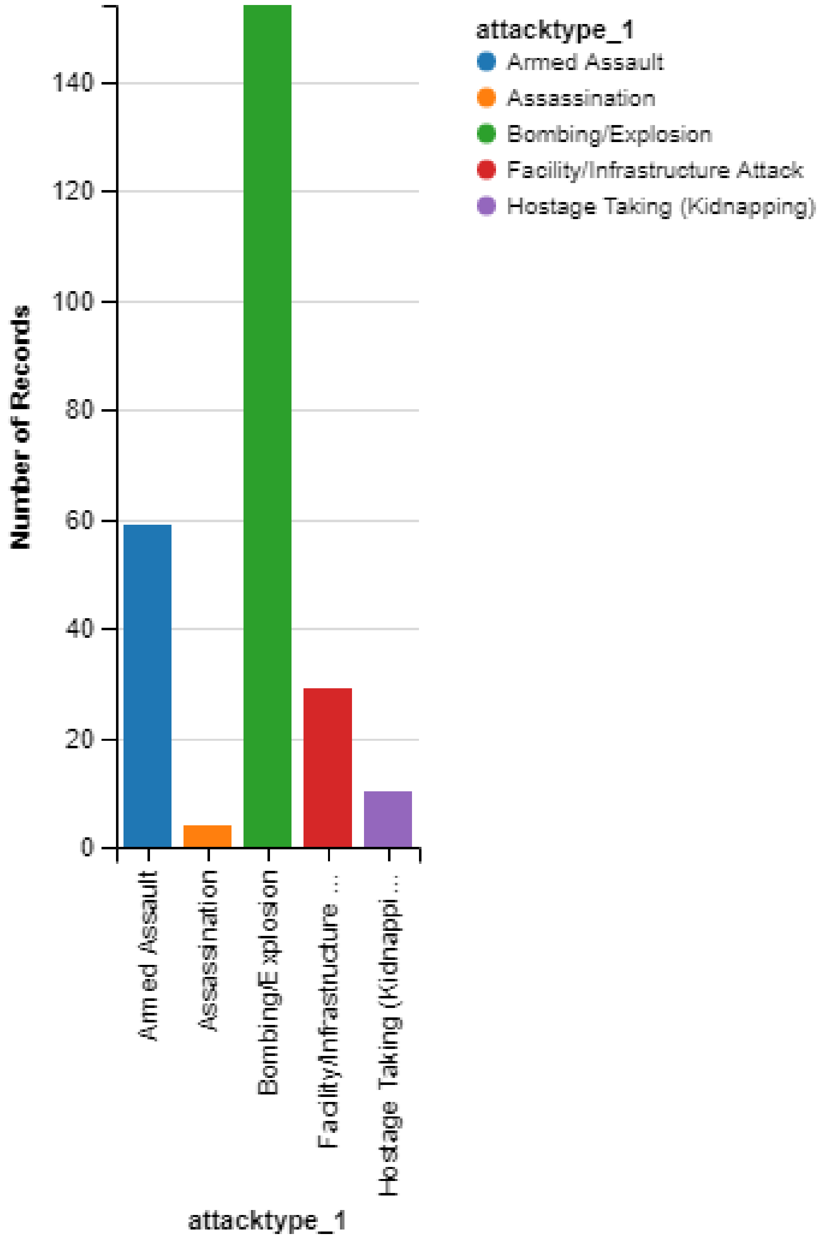
In [69]:

```
alt.Chart(top5_attacktype).mark_line().encode(
    x='month:N',
    y='count(*)',
    color='attacktype_1',
    size='count(*)'
)

# It looks like most of the attacks happened in august
```



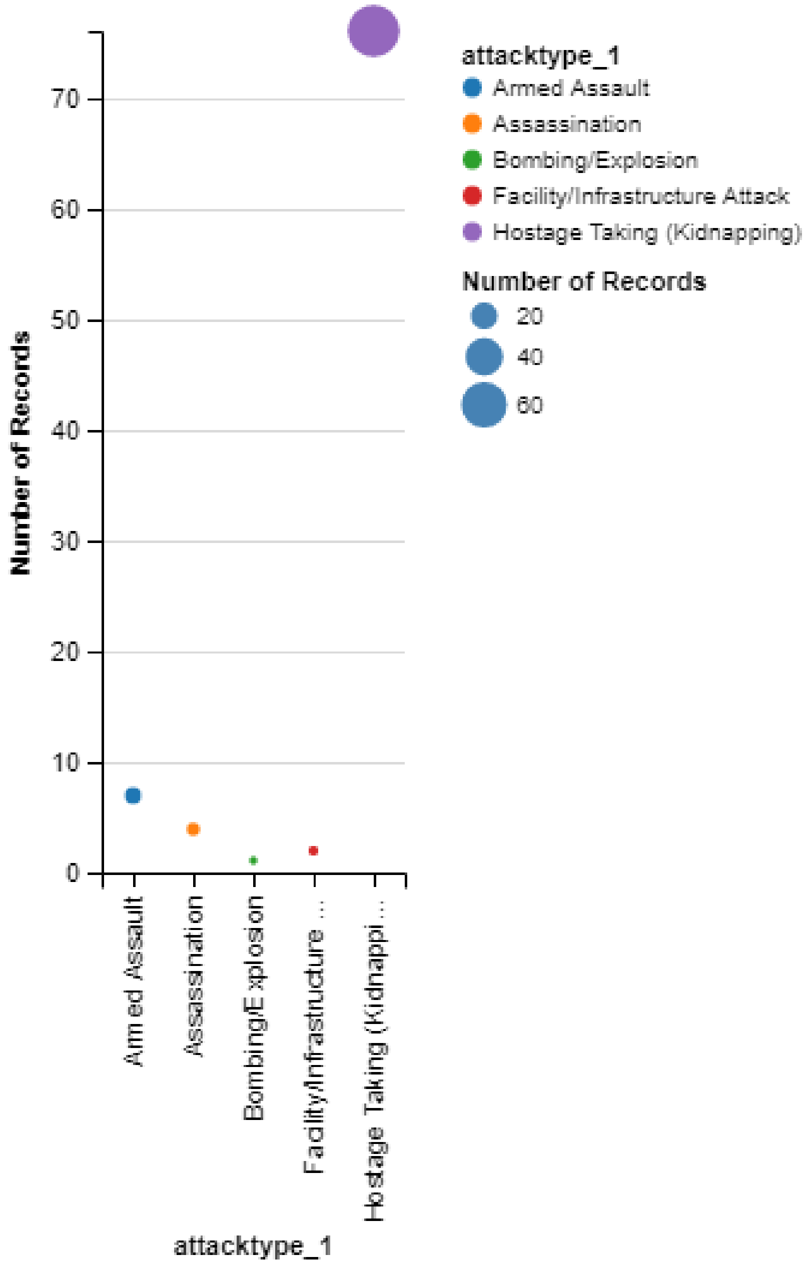
In [70]:

```
alt.Chart(top5_attacktype[top5_attacktype.multiple=='Yes']).mark_bar().enco
    x='attacktype_1',
    y='count(*)',
    color='attacktype_1',
)

# As we can see the Boming is in the most used attack type
```



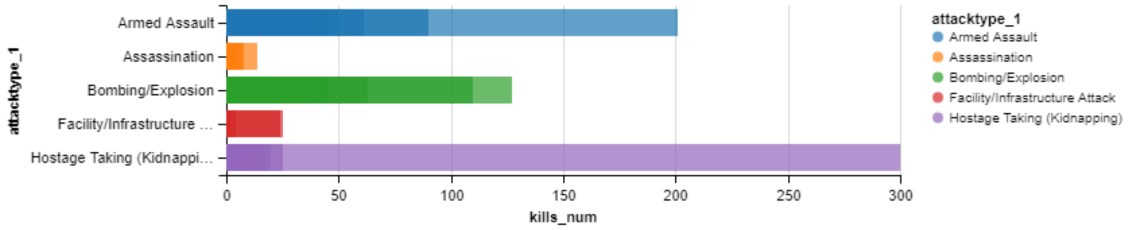
In [71]:

```
alt.Chart(top5_attacktype[top5_attacktype.extended=='Yes']).mark_circle().e
    x='attacktype_1',
    y='count(*)',
    color='attacktype_1',
    size='count(*)'
)
# The Kidnapping attack type looks like the most frequent one used in the e
```



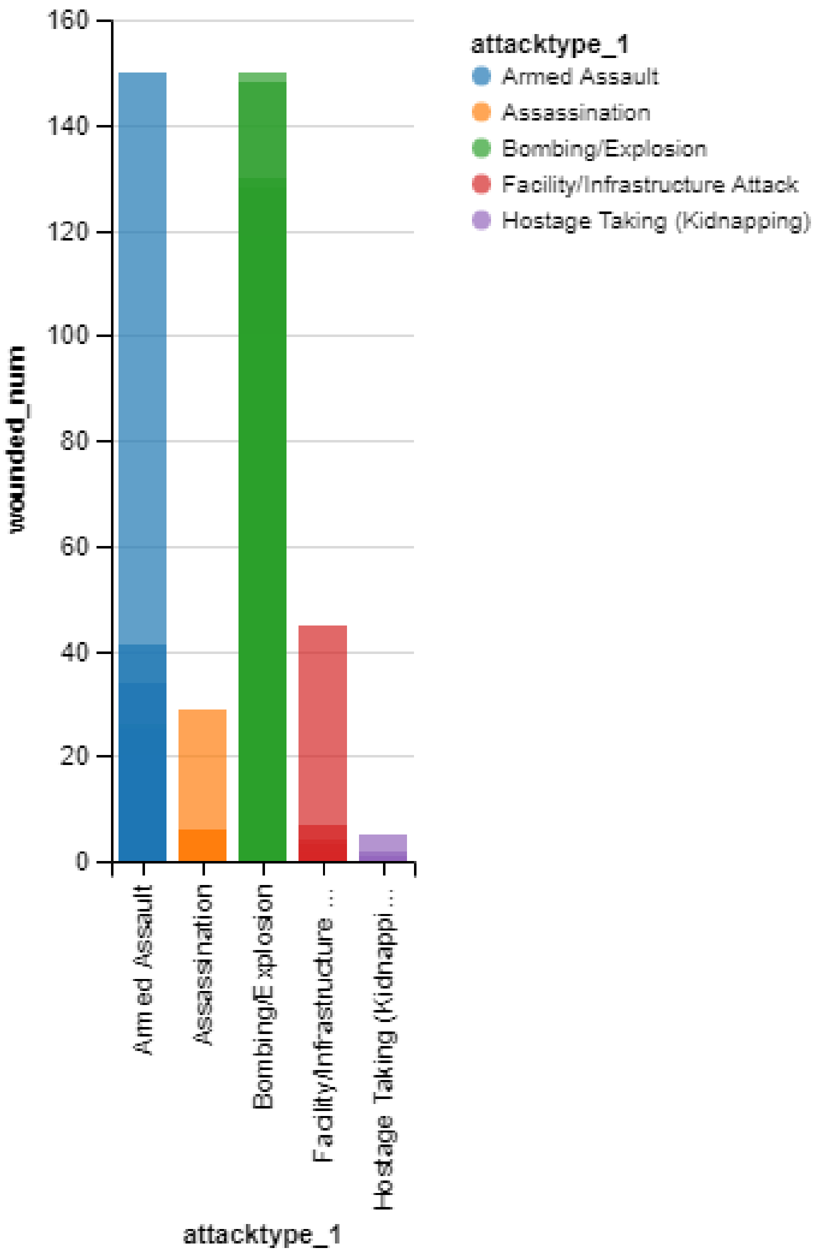
In [124]:

```
alt.Chart(top5_attacktype[top5_attacktype.kills_num.notnull()]).mark_bar().
    x='kills_num',
    y='attacktype_1',
    color='attacktype_1',
)

#Kidnapping got the highest kills number than bombing
```



In [73]:

```
alt.Chart(top5_attacktype[top5_attacktype.wounded_num.notnull()]).mark_bar(
    x='attacktype_1',
    y='wounded_num',
    color='attacktype_1',
)

# Bombing and Armed Assault got the highest wounded number of people
```



In [74]:

```
print(df_sample.weaptype[(df_sample.weaptype.notnull())&(df_sample.weaptype
```

```
Explosives/Bombs/Dynamite    986
Firearms                     651
Incendiary                   134
Melee                         43
Chemical                       4
Name: weaptype, dtype: int64
```

In [75]:

```
print(df_sample.weaptype[(df_sample.weaptype.notnull())&(df_sample.weaptype

top5_weapon_type=df_sample[df_sample.weaptype.isin([
    'Explosives/Bombs/Dynamite',
    'Firearms',
    'Incendiary',
    'Melee',
    'Chemical'])]
```

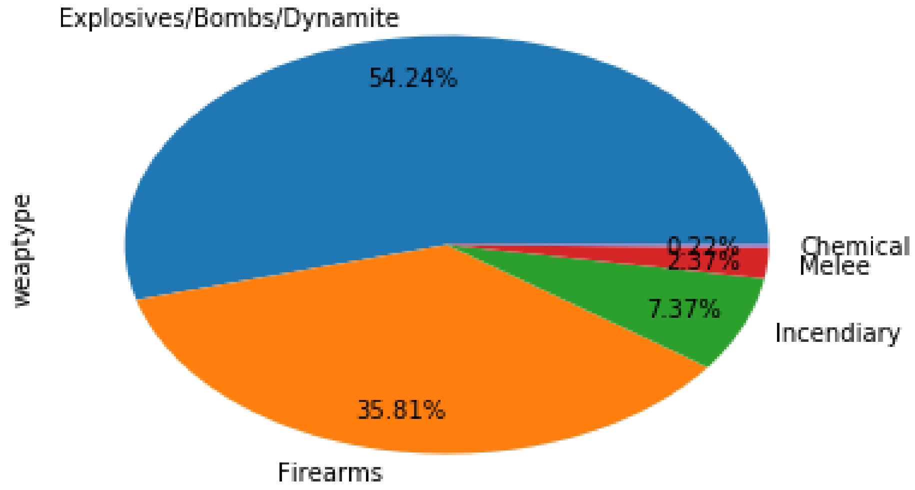
```
Explosives/Bombs/Dynamite    986
Firearms                     651
Incendiary                   134
Melee                         43
Chemical                       4
Name: weaptype, dtype: int64
```

In [76]:

```
top5_weapon_type.weaptype.value_counts().plot(kind='pie',autopct='%1.2f%%',
```

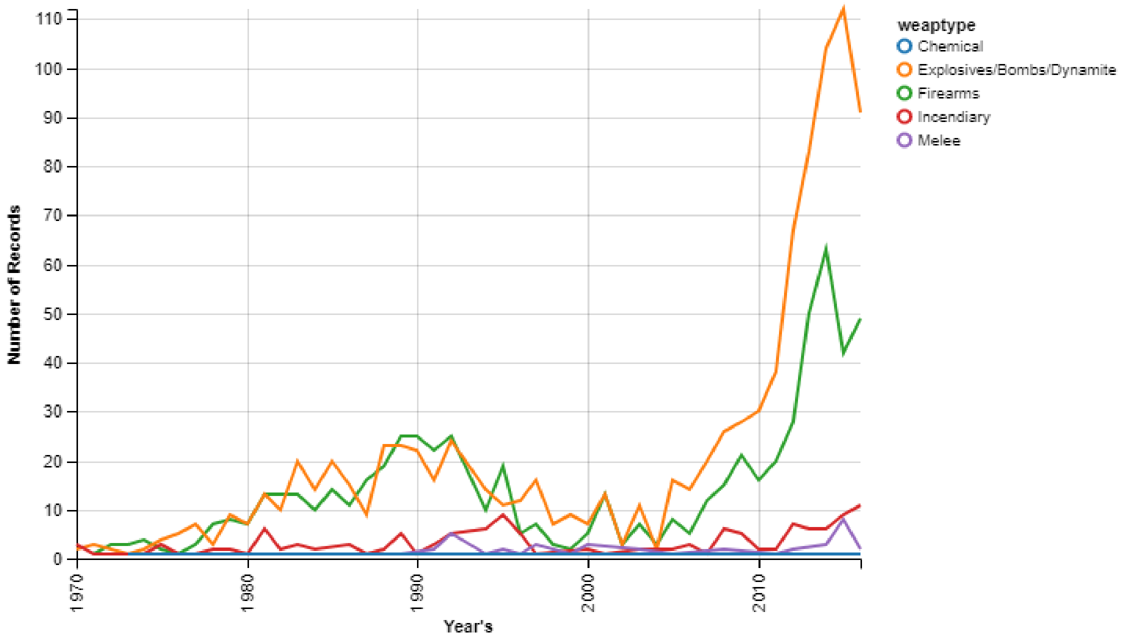
Out[76]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99c4583c8>
```



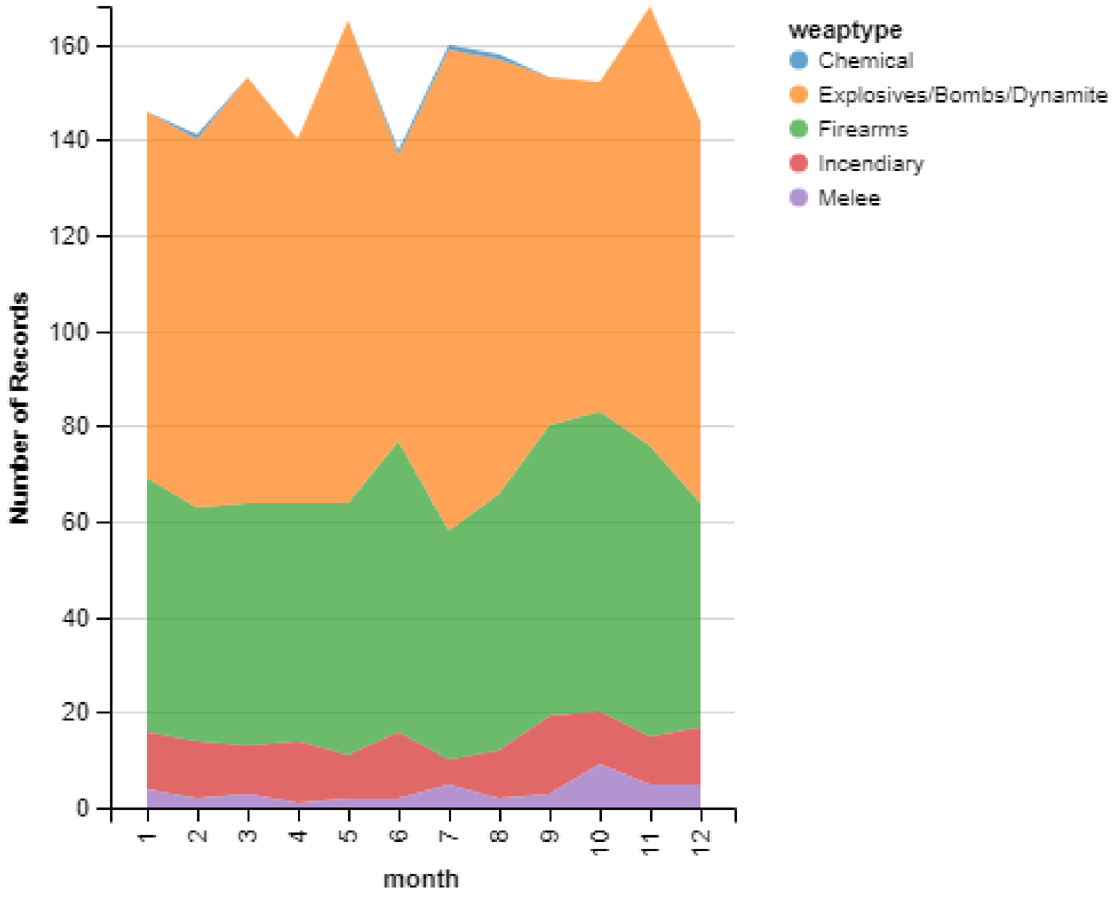
In [77]:

```
alt.Chart(top5_weapon_type).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y='count(*)',
    color='weaptype'
)
```



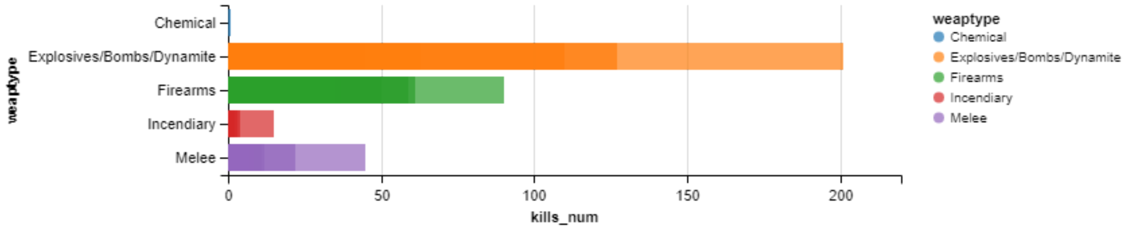
In [78]:

```
alt.Chart(top5_weapon_type).mark_area().encode(
    x='month:N',
    y='count(*)',
    color='weaptype',
    size='count(*)'
)

#Explosives mostly used in november
```



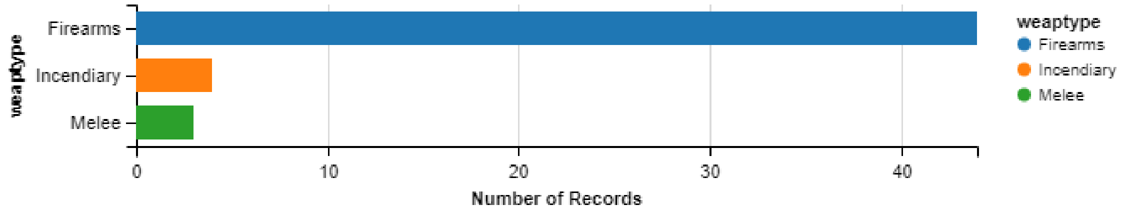
In [80]:

```
alt.Chart(top5_weapon_type[top5_weapon_type.kills_num.notnull()]).mark_bar(
    x='kills_num',
    y='weaptype',
    color='weaptype',
)

# Explosives/Bombs/Dynamite are the most weapon type used in the event's
```



In [81]:

```
alt.Chart(top5_weapon_type[top5_weapon_type.extended=='Yes']).mark_bar().en
    x='count(*)',
    y='weaptype',
    color='weaptype',
)
# FireArms are the most type who keeps the event for more than 24 hours
```



In [82]:

```
print(df_sample.targ_type[(df_sample.targ_type.notnull())&(df_sample.targ_t

top5_target_type=df_sample[df_sample.targ_type.isin([
    'Private Citizens & Property',
    'Military',
    'Police',
    'Business',
    'Government (General)'])]
```

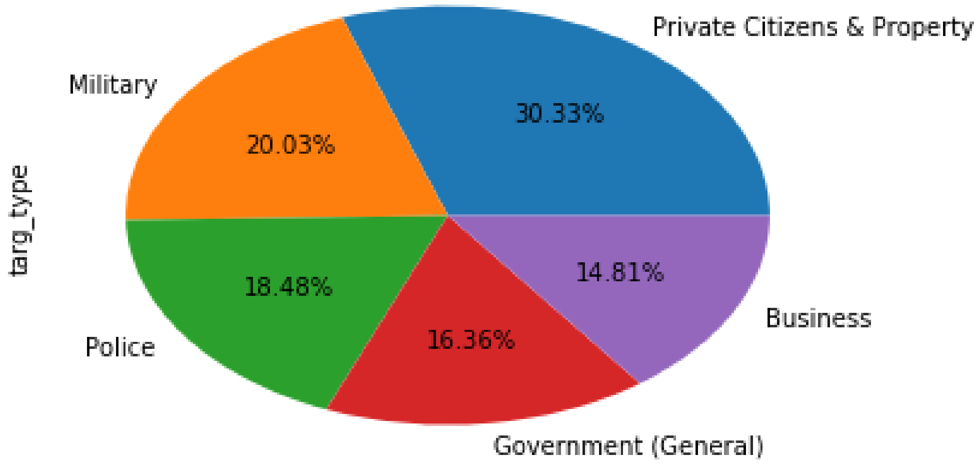
```
Private Citizens & Property    471
Military                       311
Police                         287
Government (General)           254
Business                       230
Name: targ_type, dtype: int64
```

In [83]:

```
top5_target_type.targ_type.value_counts().plot(kind='pie',autopct='%1.2f%%'
# most attacks target type are Private Citizens & Property
```

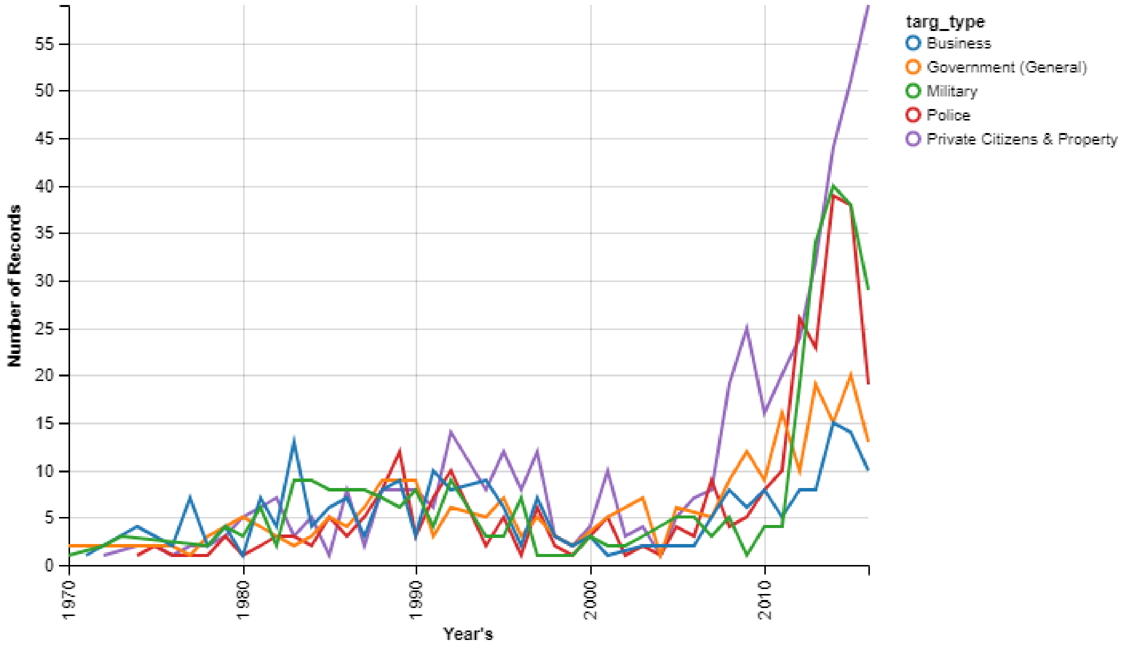
Out[83]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99c47a278>
```



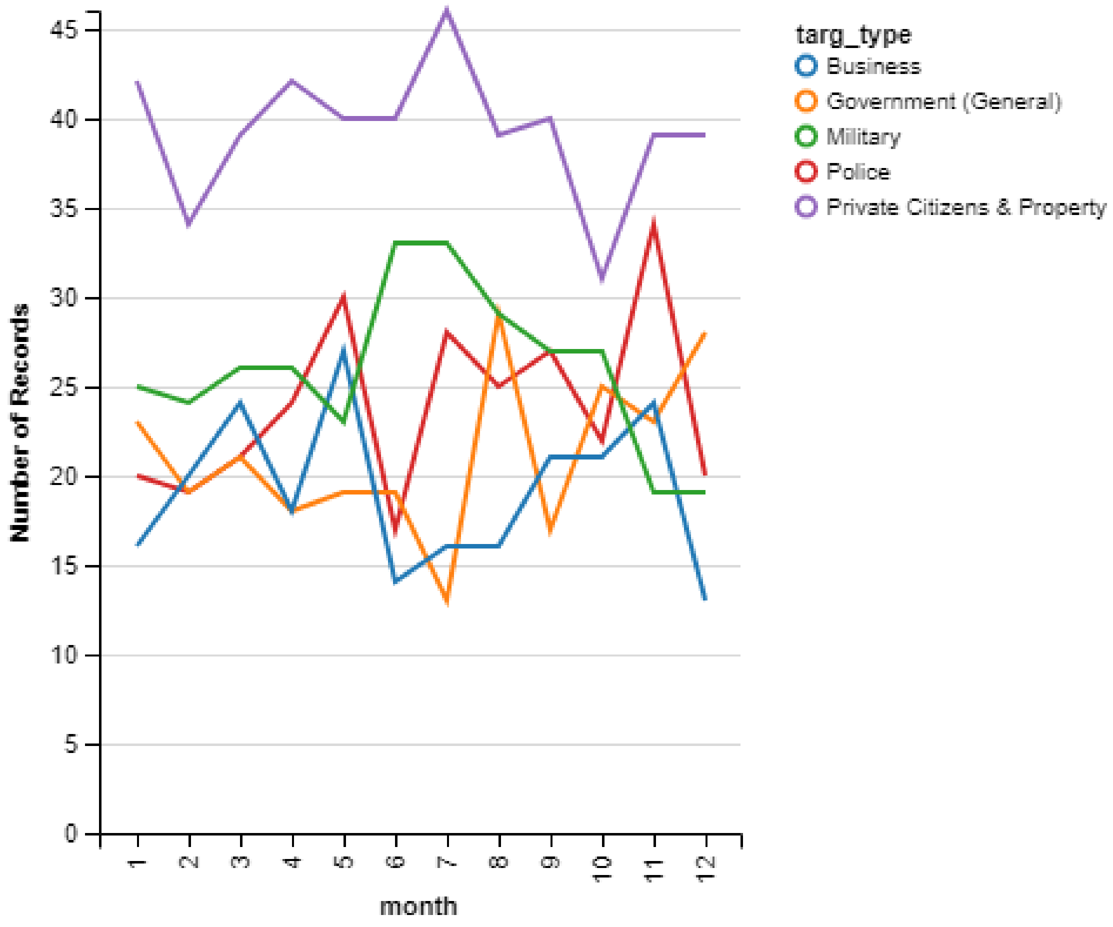
In [84]:

```
alt.Chart(top5_target_type).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y='count(*)',
    color='targ_type'
)
```



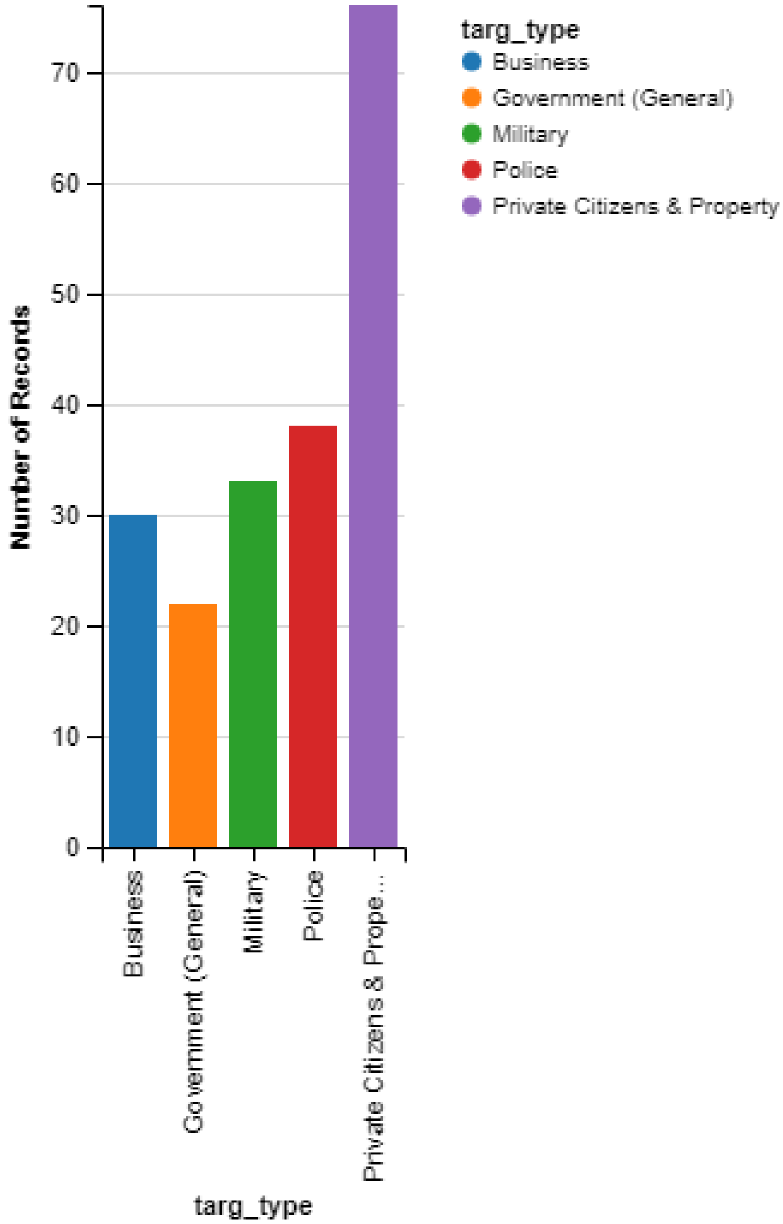
In [85]:

```
alt.Chart(top5_target_type).mark_line().encode(
    x='month:N',
    y='count(*)',
    color='targ_type',
    size='count(*)'
)

# We can see that most targeted types are private citizens
```



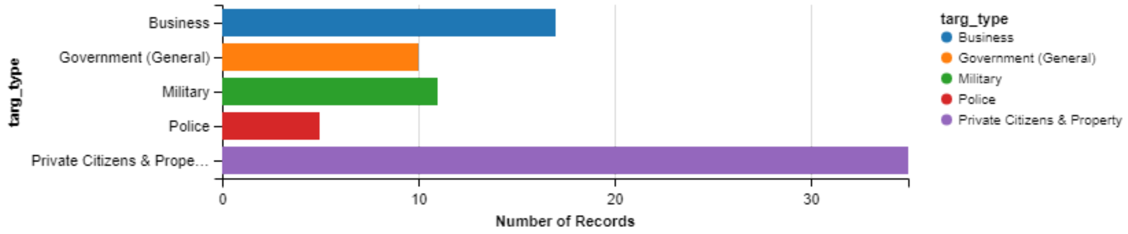
In [86]:

```
alt.Chart(top5_target_type[top5_target_type.multiple=='Yes']).mark_bar().en
    x='targ_type',
    y='count(*)',
    color='targ_type',
)

# The most multiple event's happened for targeting Private Citizen's & Prop
```



In [87]:

```
alt.Chart(top5_target_type[top5_target_type.extended=='Yes']).mark_bar().en
    x='count(*)',
    y='targ_type',
    color='targ_type',
)

# event's extended for more than 24 hours usually have something to do with
# 1st (Private Citizen's & Property)
# 2nd Business
```



In [88]:

```
print(df_sample.targ_subtype[
    (df_sample.targ_subtype.notnull())&(df_sample.targ_subtype!='Unnamed Ci

#Here we can see the Sub type target
```

```
Police Security Forces/Officers                            14
1
Military Unit/Patrol/Convoy                                 9
9
Military Personnel (soldiers, troops, officers, forces)     8
3
Government Personnel (excluding police, military)           7
8
Politician or Political Party Movement/Meeting/Rally        6
9
Name: targ_subtype, dtype: int64
```

In [89]:

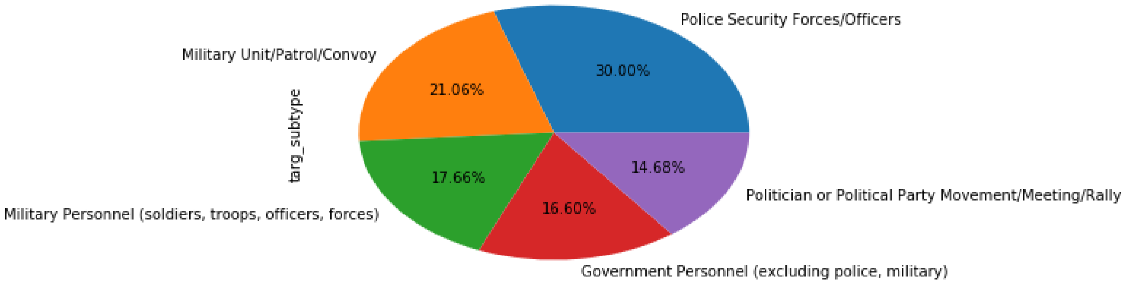
```
top5_target_Subtype=df_sample[df_sample.targ_subtype.isin([
    'Police Security Forces/Officers',
    'Military Unit/Patrol/Convoy',
    'Military Personnel (soldiers, troops, officers, forces)',
    'Government Personnel (excluding police, military)',
    'Politician or Political Party Movement/Meeting/Rally'])]
```

In [90]:

```
top5_target_Subtype.targ_subtype.value_counts().plot(kind='pie',autopct='%1
```

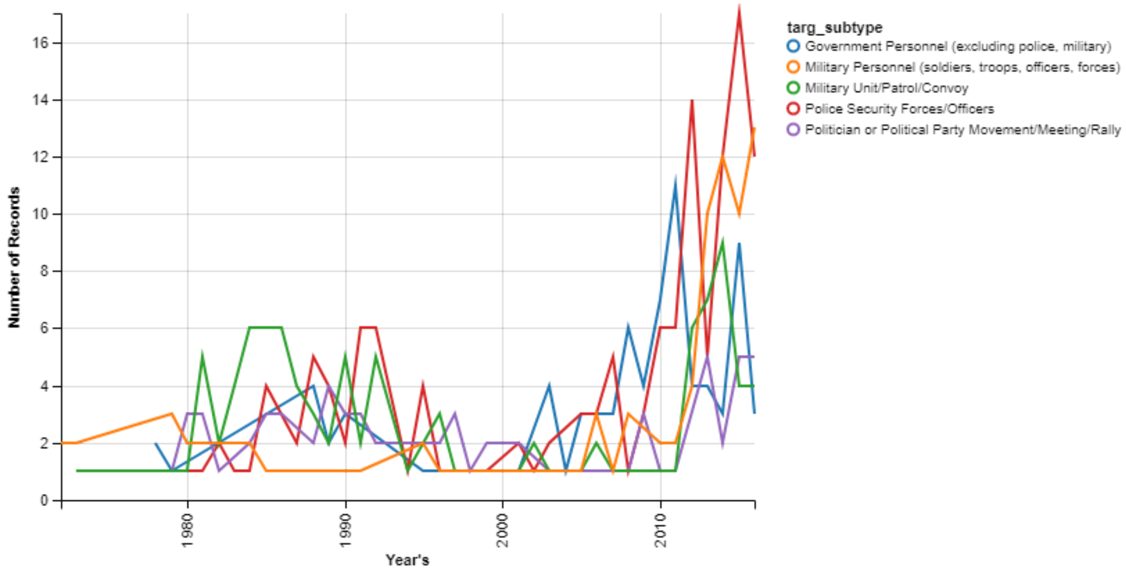
Out[90]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1d99b552518>
```



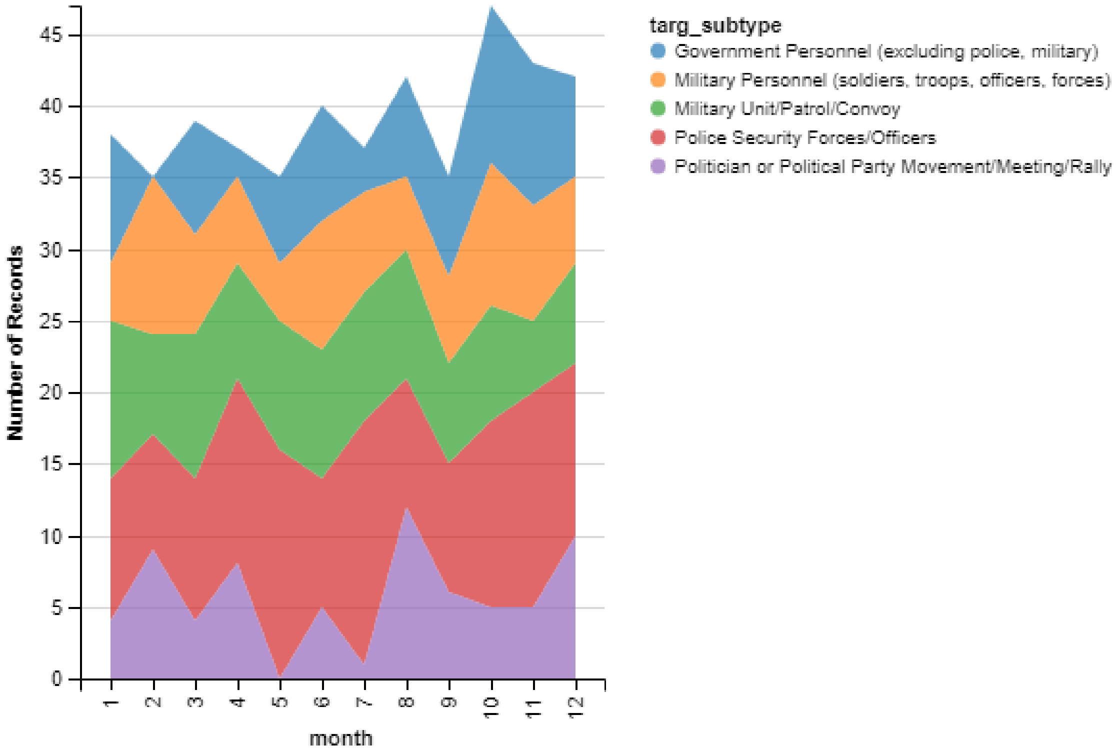
In [91]:

```
alt.Chart(top5_target_Subtype).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y='count(*)',
    color='targ_subtype'
)
```



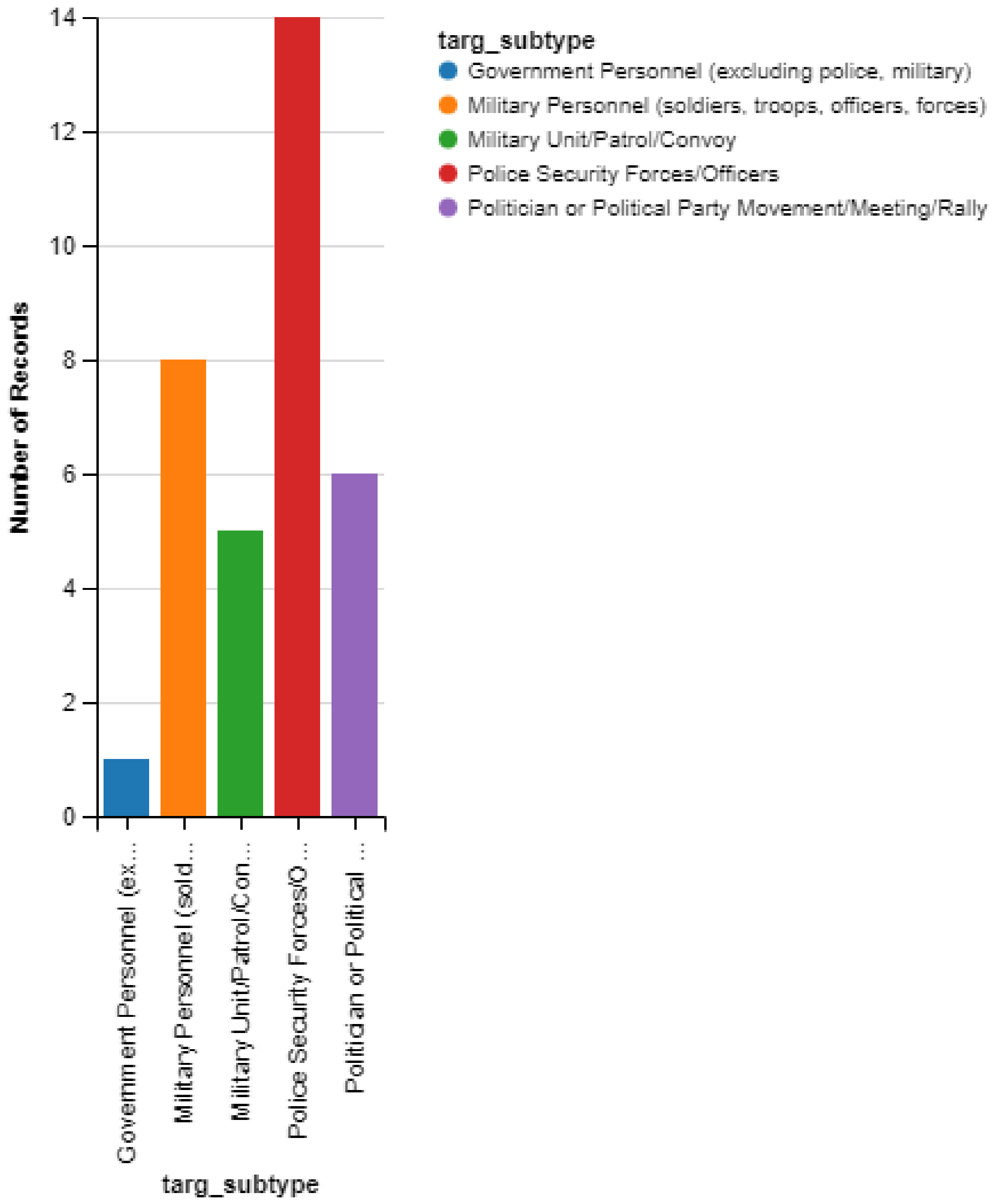
In [92]:

```
alt.Chart(top5_target_Subtype).mark_area().encode(
    x='month:N',
    y='count(*)',
    color='targ_subtype',
    size='count(*)'
)
```



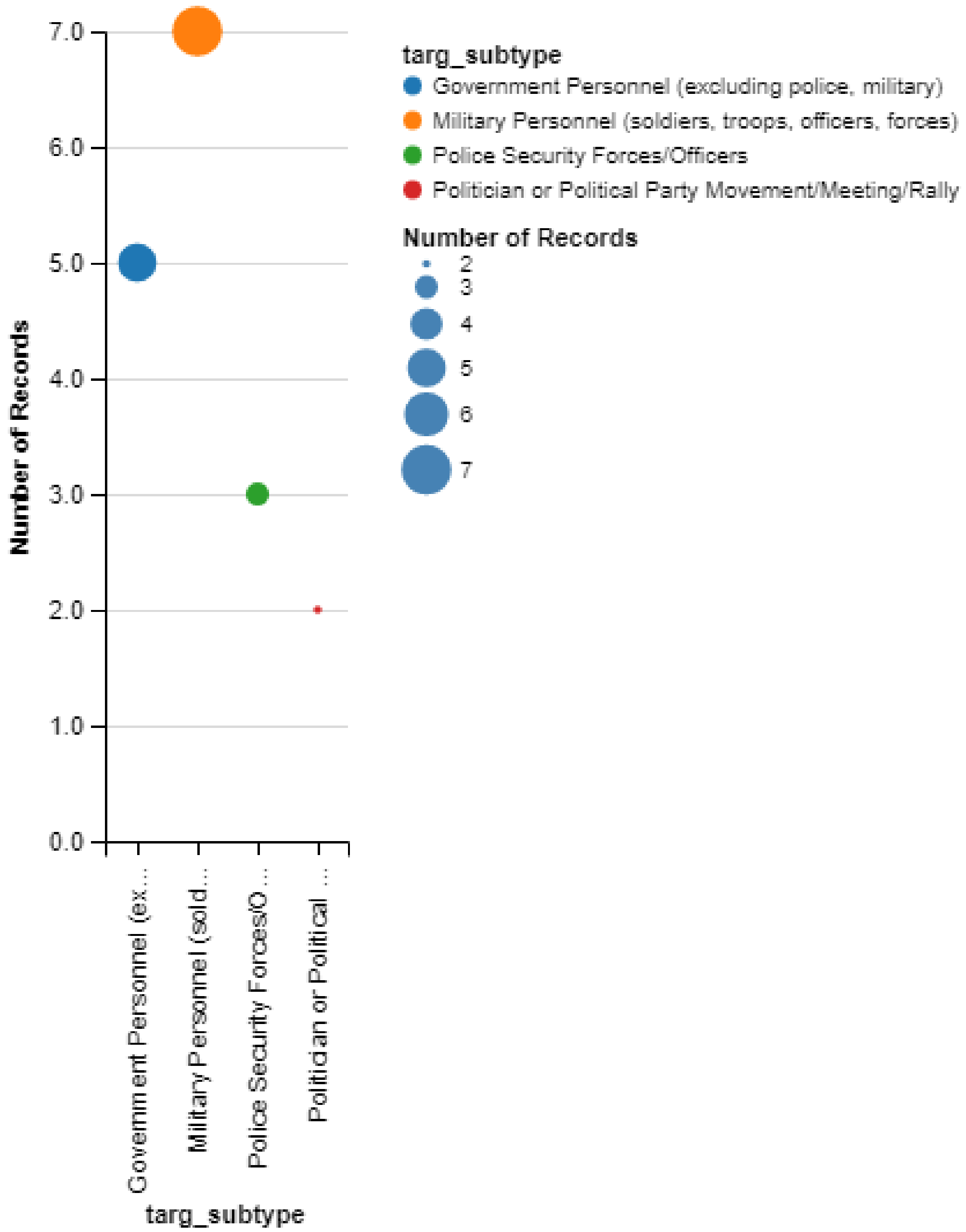
```
In [93]:
```

```
alt.Chart(top5_target_Subtype[top5_target_Subtype.multiple=='Yes']).mark_ba
    x='targ_subtype',
    y='count(*)',
    color='targ_subtype',
)

#The following subtype are more likely to get into multiple events:
# 1st Police Security Forces/Officers
# 2nd Military Personnel (soldiers, troops, officers, forces)
```



In [94]:

```
alt.Chart(top5_target_Subtype[top5_target_Subtype.extended=='Yes']).mark_ci
    x='targ_subtype',
    y='count(*)',
    color='targ_subtype',
    size='count(*)'
)

# The most subtype who gets into event's stays longer than 24 hour's are
# the Government Personnel (excluding police, military)
```



In [6]:

```
print(df_sample.group_name[
    (df_sample.group_name.notnull())&(df_sample.group_name!='Unknown')].val

#Here we can see the Sub type target
```

```
...
```

In [7]:

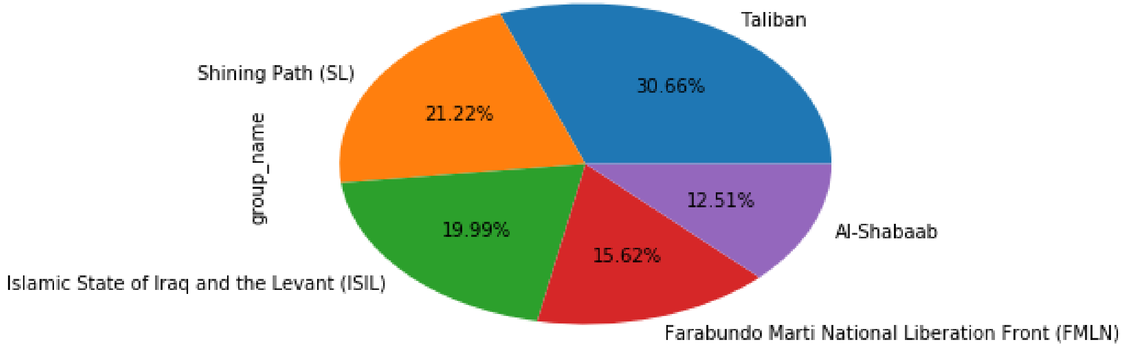
```
top5_target_grpName=df_sample[df_sample.group_name.isin([
    'Taliban',
    'Shining Path (SL)',
    'Islamic State of Iraq and the Levant (ISIL)',
    'Farabundo Marti National Liberation Front (FMLN)',
    'Irish Republican Army (IRA)'])]
```

In [16]:

```
top5_target_grpName.group_name.value_counts().plot(kind='pie',autopct='%1.2
```

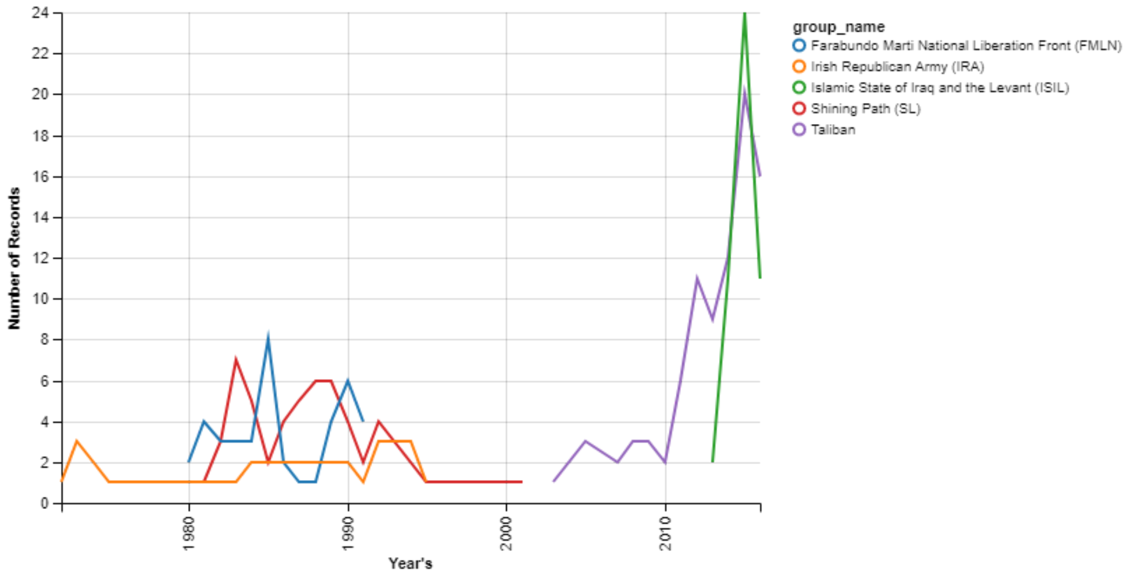
Out[16]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x2b2d68150f0>
```



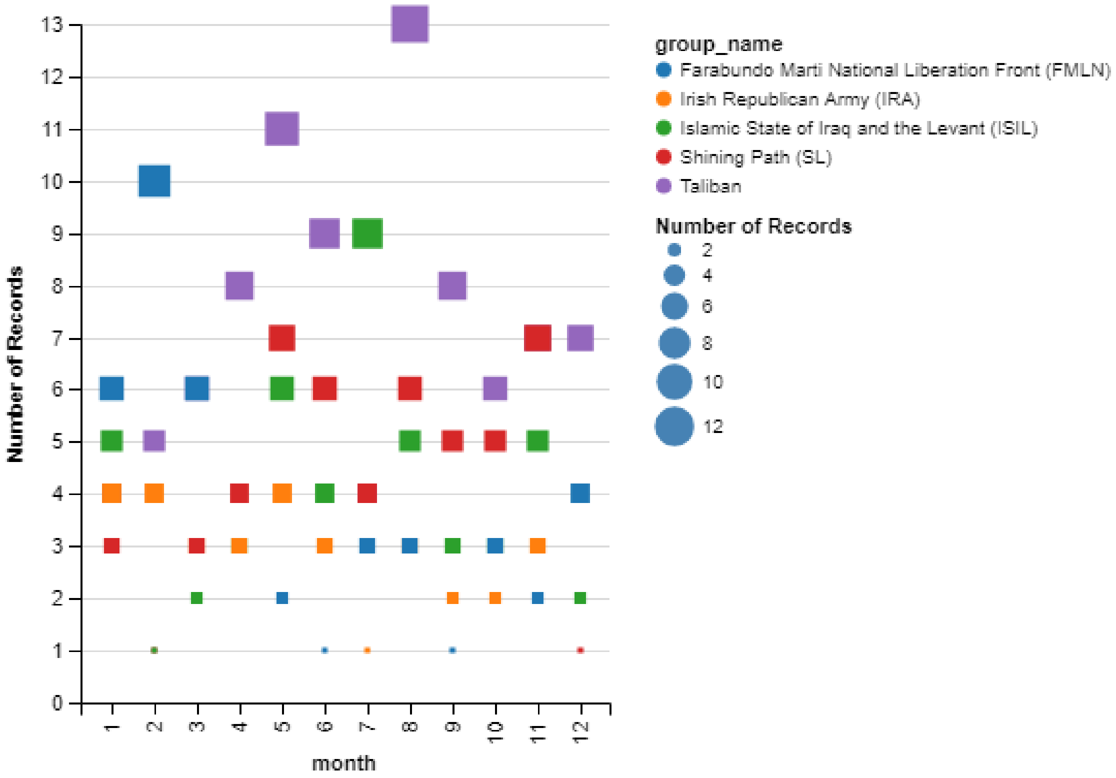
In [98]:

```
alt.Chart(top5_target_grpName).mark_line().encode(
    x=alt.X('year:T', timeUnit='year',title="Year's"),
    y='count(*)',
    color='group_name'
)

# we can see that the ISIL and Taliban the most active groups
```



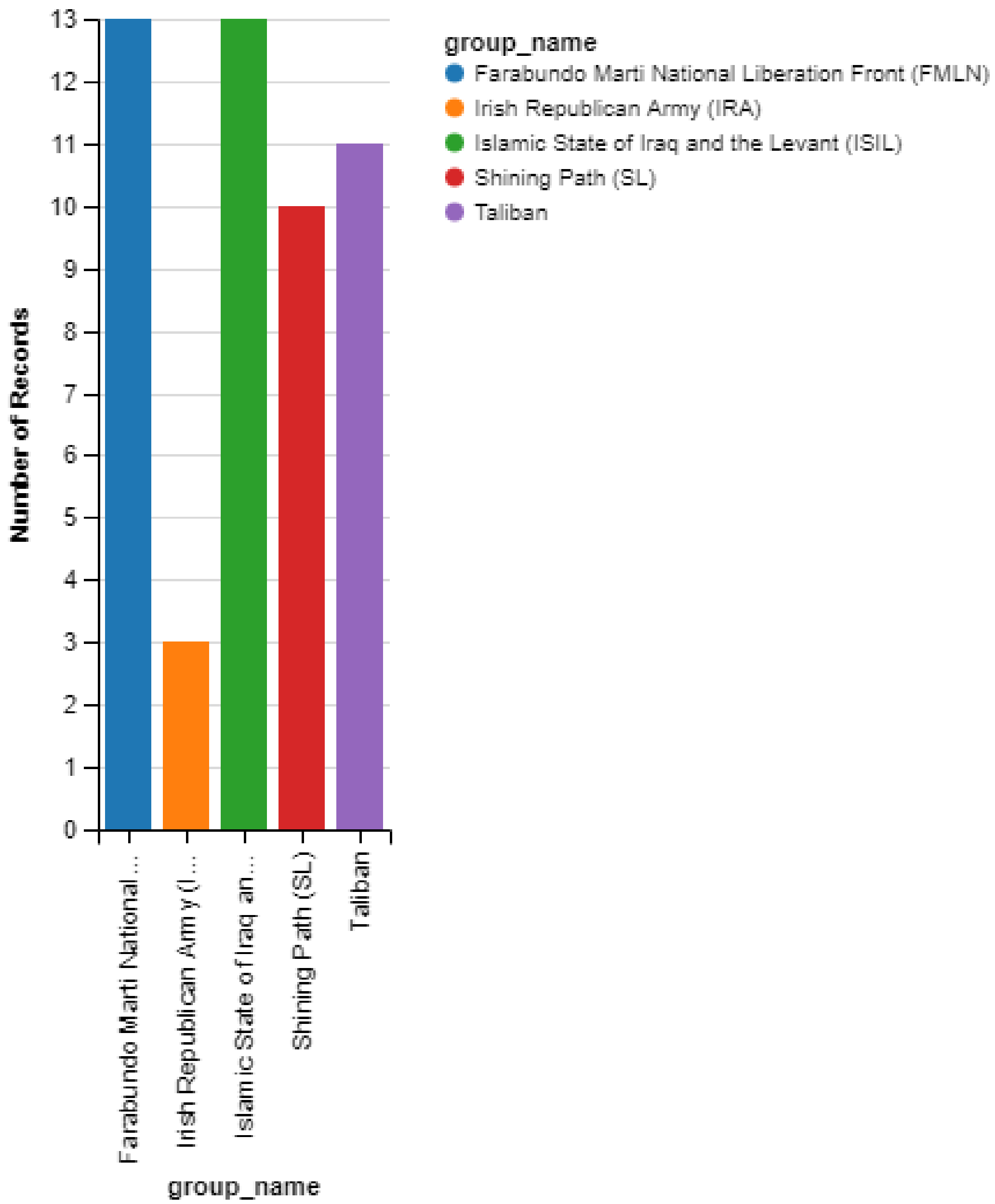
In [99]:

```
alt.Chart(top5_target_grpName).mark_square().encode(
    x='month:N',
    y='count(*)',
    color='group_name',
    size='count(*)'
)

# We can see the groups activities got lower in the beginning and the end o
```



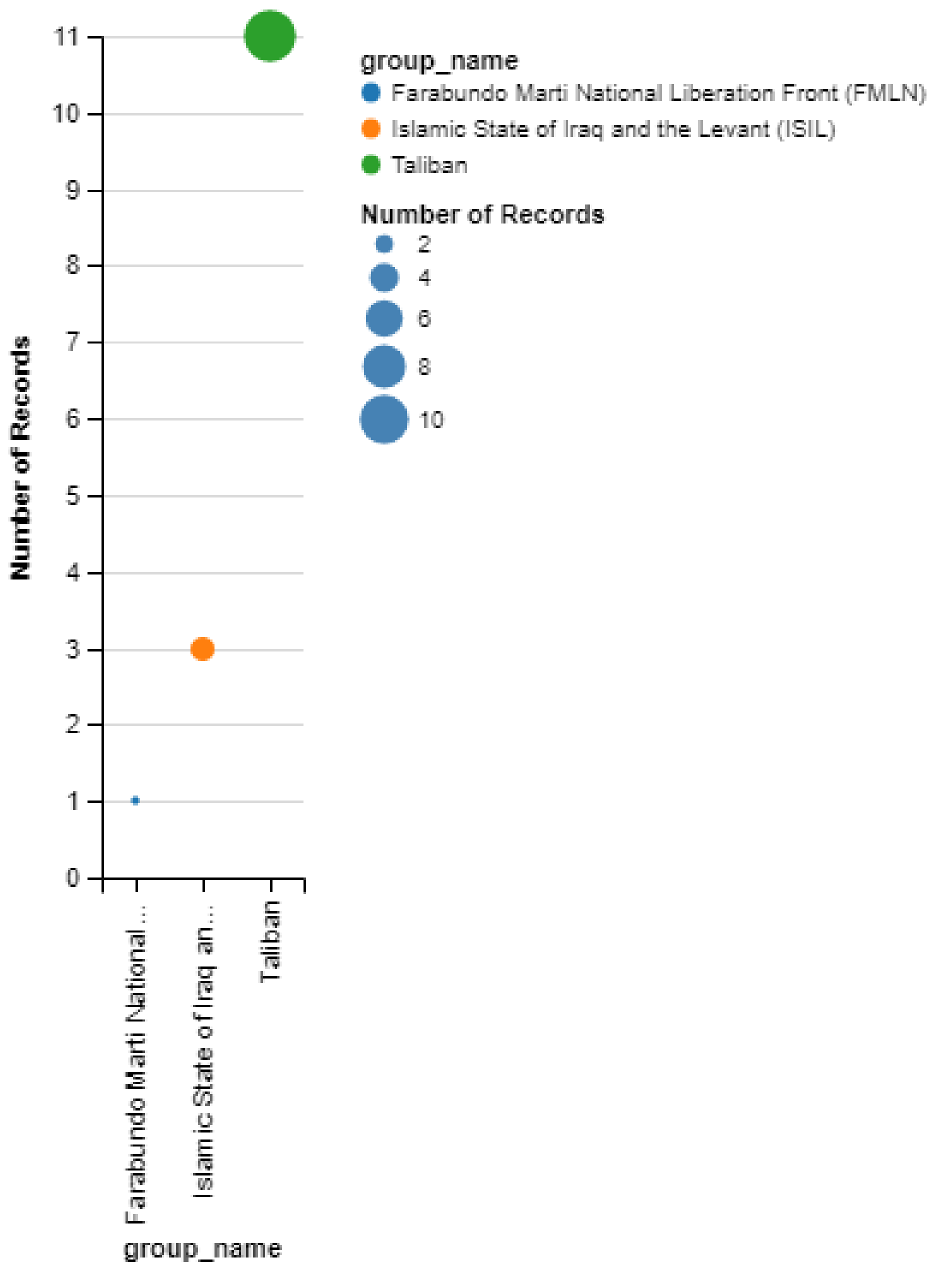
In [100]:

```
alt.Chart(top5_target_grpName[top5_target_grpName.multiple=='Yes']).mark_ba
    x='group_name',
    y='count(*)',
    color='group_name',
)

#Here we can see usually ISIL cause multiple event's
```



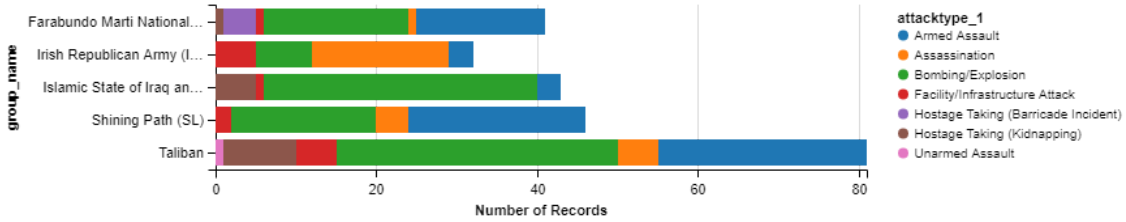
In [101]:

```
alt.Chart(top5_target_grpName[top5_target_grpName.extended=='Yes']).mark_ci
    x='group_name',
    y='count(*)',
    color='group_name',
    size='count(*)'
)


# Groups who get their event's to extend for more than 24 hour's:
#1st Taliban
#2nd ISIL
```



In [102]:

```
alt.Chart(top5_target_grpName[top5_target_grpName.attacktype_1!='Unknown'])
    x='count(*)',
    y='group_name',
    color='attacktype_1',
)

#The most used attack type in Taliban is Boming and Armed Assault
```


group_name
Farabundo Marti National...
Irish Republican Army (I...
Islamic State of Iraq an...
Shining Path (SL)
Taliban
Number of Records
attacktype_1
● Armed Assault
● Assassination
● Bombing/Explosion
● Facility/Infrastructure Attack
● Hostage Taking (Barricade Incident)
● Hostage Taking (Kidnapping)
● Unarmed Assault